Safe and Secure Control of Connected and Automated Vehicles

by

Mohammad Hossein Basiri

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Electrical and Computer Engineering

Waterloo, Ontario, Canada, 2020

Examining Committee Membership

The following served on the Examining Committee for this thesis. The decision of the Examining Committee is by majority vote.

External Examiner:	Prof. Fengjun Yan Associate Professor, Dept. of Mechanical Engineering McMaster University
Supervisors:	Prof. Sebastian Fischmeister Professor, Dept. of Electrical and Computer Engineering, University of Waterloo Prof. Nasser Lashgarian Azad Associate Professor, Dept. of Systems Design Engineering, University of Waterloo
Internal Member:	Prof. David Wang Professor, Dept. of Electrical and Computer Engineering, University of Waterloo Prof. Daniel Miller Professor, Dept. of Electrical and Computer Engineering, University of Waterloo
Internal-External Member:	Prof. Amir Khajepour Professor, Dept. of Mechanical and Mechatronics Engineering, University of Waterloo

Authors Declaration

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

- **M. H. Basiri**, N. L. Azad, and S. Fischmeister, "Secure Dynamic Nonlinear Heterogeneous Vehicle Platooning: Denial-of-Service Cyber-Attack Case," *In: Security and Privacy in Cyber-Physical Systems: Threats and Defenses. Studies in Systems, Decision and Control, In Press. Springer, 2021.*

- **M. H. Basiri**, M. Pirani, N. L. Azad, and S. Fischmeister, "Security-Aware Optimal Actuator Placement in Vehicle Platooning," *Asian Journal of Control, 2020.*

  Contribution of M. Pirani: Collaboration in the development of the actuator placement strategy.

- **M. H. Basiri**, N. L. Azad, and S. Fischmeister, "Attack Resilient Heterogeneous Vehicle Platooning Using Secure Distributed Nonlinear Model Predictive Control," *28th Mediterranean Conference on Control and Automation (MED), Saint-Raphael, France, IEEE, 2020.*

- **M. H. Basiri**, B. Ghojogh, N. L. Azad, S. Fischmeister, F. Karray, and M. Crowley, "Distributed Nonlinear Model Predictive Control and Metric Learning for Heterogeneous Vehicle Platooning with Cut-in/Cut-out Maneuvers," *59th Conference on Decision and Control (CDC), Jeju Island, Republic of Korea, 2020.*

  Contribution of B. Ghojogh: Collaboration in the metric learning analyses.

- **M. H. Basiri**, M. Pirani, N. L. Azad, and S. Fischmeister, "Security of Vehicle Platooning: A Game-Theoretic Approach," *IEEE Access, 185565-185579, 2019.*

  Contribution of M. Pirani: Collaboration in the development of the sensor placement strategy.

Abstract

Evolution of Connected and Automated Vehicles (CAV), as an important class of Cyber-Physical Systems (CPS), plays a crucial role in providing innovative services in transport and traffic management. *Vehicle platoons*, as a set of CAV, forming a string of connected vehicles, have offered significant enhancements in traffic management, energy consumption, and safety in intelligent transportation systems. However, due to the existence of the cyber layer in these systems, subtle security related issues have been underlined and need to be taken into account with sufficient attention. In fact, despite the benefits brought by the platoons, they potentially suffer from insecure networks which provide the connectivity among the vehicles participating in the platoon which makes these systems prone to be under the risk of cyber attacks. One (or more) external intelligent intruder(s) might attack one (or more) of the vehicles participating in a platoon. In this respect, the need for a safe and secure driving experience is highly sensible and crucial. Hence, we will concentrate on improving the safety and security of CAVs in different scenarios by taking advantage of security related approaches and CAV control systems.

In this thesis, we are going to focus on two main levels of platoon control, namely I) High level secure platoon control, and II) Low level secure platoon control. In particular, in the high level part, we consider platoons with arbitrary inter-vehicular communication topoloy whereby the vehicles are able to exchange their driving data with each other through DSRC-based environment. The whole platoon is modeled using graph-theoretic notions by denoting the vehicles as the nodes and the inter-vehicular communication quality as the edge weights. We study the security of the vehicle platoon exposed to cyber attacks using a novel game-theoretic approach. The platoon topologies under investigation are directed (called predecessor following) or undirected (bidirectional) weighted graphs. The attacker-detector game is defined as follows. The attacker targets some vehicles in the platoon to attack and the detector deploys monitoring sensors on the vehicles. The attacker's objective is to be as stealthy to the sensors as possible while the detector tries to place the monitoring sensors to detect the attack impact as much as he can. The existence of equilibrium strategies for this game is investigated based on which the detector can choose specific vehicles to put his sensors on and increase the security level of the system. Moreover, we study the effect of adding (or removing) communication links between vehicles on the game value. We then address the same problem while investigating the optimal actuator placement strategy needed by the defender to mitigate the effects of the attack. In this respect, the energy needed by the attacker to steer the consensus follower-leader dynamics of the system towards his desired direction is used as the game payoff. Simulation and experimental results conducted on a vehicle platoon setup using Robotic Operating System (ROS) demonstrate the effectiveness of our analyses.

In the low level platoon control, we exploit novel secure model predictive controller algorithms to provide suitable countermeasure against a prevalent data availability attack, namely Denial-of-Service (DoS) attack. A DoS intruder can endanger the security of platoon by jamming the communication network among the vehicles which is responsible to transmit inter-vehicular data throughout the platoon. In other words, he may cause a failure in the network by jamming it or injecting a huge amount of delay, which in essence makes the outdated transferred data useless. This can potentially result in huge performance degradation or even hazardous collisions. We propose novel secure distributed nonlinear model predictive control algorithms for both static and dynamic nonlinear heterogeneous platoons which are capable of handling DoS attack performed on a platoon equipped by different communication topologies and at the same time they guarantee the desired formation control performance. Notably, in the dynamic case, our proposed method is capable of providing safe and secure control of the platoon in which arbitrary vehicles might perform cut-in and/or cut-out maneuvers. Convergence time analysis of the system are also investigated. Simulation results on a sample heterogeneous attacked platoon exploiting two-predecessor follower communication environment demonstrates the fruitfulness of the method.

Acknowledgements

First and foremost, I would like to express my many thanks to Prof. Sebastian Fischmeister and Prof. Nasser Lashgarian Azad, for their endless support, encouragement, and supervision of the research presented in this thesis.

I extend my appreciation to Prof. Fengjun Yan (McMaster University), Prof. Amir Khajepour, Prof. Daniel Miller, and Prof. David Wang for taking time to be in my committee and guide me with their helpful comments.

I want to express my appreciation to Prof. John. G. Thistle, Prof. John W. Simpson-Porco (University of Toronto), Dr. Mohammad Pirani, and Dr. Yang Zheng (Harvard University) for the interactions and scientific collaborations that we have had which inspired me in my research.

I am also grateful to all of my friends in Waterloo and Toronto for all the hangouts, laughter, and travels that we have shared.

Most importantly, none of this would have been possible without the patience, love and support of my parents, whom I cordially dedicate the honor of all my current and future achievements to.

Dedication

To my parents

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Over recent years, the need to establish distributed control systems has been one of the significant motivations for emergence of large scale systems. This becomes feasible through newly developed systems, namely Cyber-Physical Systems (CPS), which provides the opportunity not only to have large scale wide spread control systems but also to take advantage of wireless data communication approaches. In other words, CPS are among the fast emerging profound infrastructures enabling traditional physical plants to operate in a wide area and a distributed fashion. Networking, computation, communication, and control are tightly interwoven to foster a CPS [1]. These components are categorized in cyber and physical layers, each of which interacts with the other parts to receive external data, process them, and generate appropriate output signals. Never may a CPS perform correctly without properly and timely functioning of its constituents. Instances of CPS include, but not limited to, Connected and Automated Vehicles (CAV), medical monitoring, robotic systems, Unmanned Aerial Vehicles (UAV), and smart grid [2]. In this thesis, we will focus on CAV in platooning as our primary application. Although, benefiting from communication channels has enhanced the solicitation for CPS, the vulnerability of such systems to malicious external attackers has attracted a great amount of attention [3]. Notably, apart from the physical layer, the cyber one has been broadly shown to be prone to external intelligent cyber attacks. Data integrity, confidentiality, and availability are the major crucial concerns of cyber-security that an intelligent intruder might target [4,5]. Various attacks have been introduced to destruct one or more of the aforementioned security aspects of a CPS. False data injection, GPS spoofing, eavesdropping, Denial-of-Service

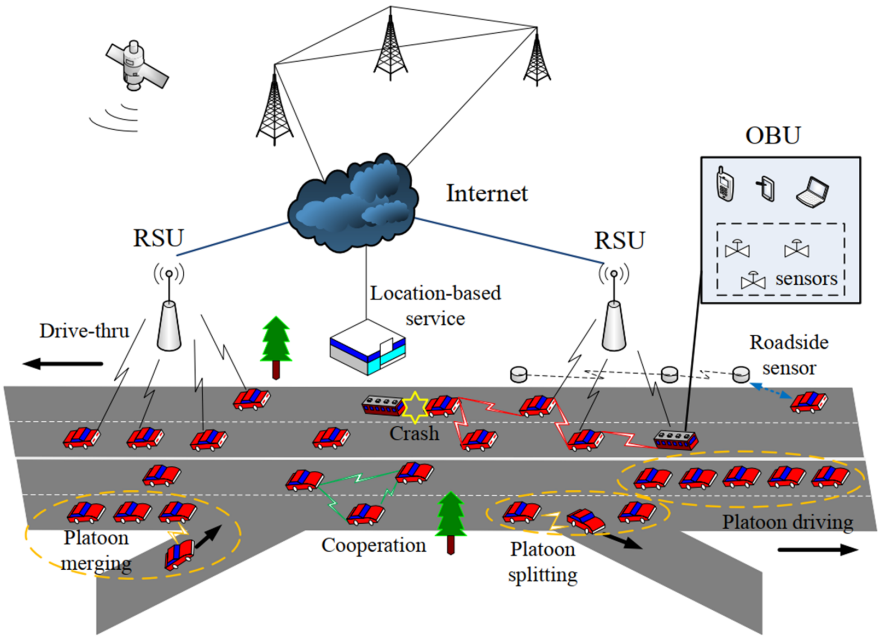


Figure 1.1: A general scheme of vehicular cyber-physical system [15]

(DoS), and replay attack are some of the paradigms [6–10]. Several well-known attacks on CPS include Stuxnet on a Supervisory Control and Data Acquisition (SCADA) system [11, 12], attacks on the wireless network channels in smart power grid systems [13], and compromising Anti-lock Braking System (ABS) sensors of a vehicle [14]. Hence, the security related issues, such as attack detection and secure state estimation, and control of CPS have been converted to attracting challenges in the control community.

CAVs are a class of CPS as they utilize communication, computation, sensing and actuation (see Fig. 1.1). CAVs can communicate with each other and exchange their date through Vehicle-to-Vehicle (V2V) and/or Vehicle-to-Infrastructure (V2I) wireless communications. We need to notice that this structure can be exposed to malicious adversarial attacks as well. Either the communication links can be in the risk of delays and/or other false data injection attacks or there might be a malicious agent among the autonomous cars. The latter case can happen in a setting in which the connected cars include several agents driving based on their own learned policies. Since we intend to focus on CAV platooning as our main application, we provide a comprehensive introductory material and some backgrounds on this application in the following.

Due to the huge growth in the number of vehicles driving in the world, traffic congestion

threaten the driving safety. This will potentially result in increasing the risk of accidents. Autonomous vehicles and autonomous driving is another aspect which has got a great deal of attention. Through this technological development, driving safety can be highly enhanced as most of the car accidents are caused by human errors and distractions while driving. Over 90% of all car accidents are caused by human errors [16]. From this point of view, self driving cars can remove a considerable amount of human errors resulting in safer transportation. In Canada alone there were close to 111,000 road related injuries, and over 1,800 fatalities reported in 2014 [17]. Autonomous cars have many other advantages such as getting faster to the destination, reducing governmental costs and car ownership [18,19].

The degree of autonomy incorporated in autonomous driving is categorized in 6 different levels (levels 0-5). Level 0 (no automation) is the most basic one in which no autonomy is incorporated. The vehicle is fully controlled by a human driver. In level 1 (driver assistance), the vehicle can assist the driver with some functions, such as steering, acceleration, or braking. In level 2 (partial automation) the vehicle lets the driver disinvolve with some of these tasks. The driver has still the main role to monitor the environment and to take care of most safety-critical functions. The driver is responsible to take the full control of the vehicle when needed. In level 3 (conditional automation), the vehicle performs all the environment monitoring tasks. In safe conditions, the driver can leave the safety-critical functions like braking to the vehicle; however, his attention is still required. Level 4 (high automation) of autonomy is able to take care of monitoring the environment, steering, acceleration, and braking. In addition, the vehicle is capable of changing lanes, turning and using signals. However, the vehicle can not perform decisions in more complex scenarios, such as traffic jams or merging onto the highway. Level 5 (complete automation) exploits a full autonomy which requires no human intervention, pedals, brakes, or a steering wheel.

As was mentioned before, autonomous cars can be equipped with wireless data communication devices such that they can transfer data such as inter-vehicular distance and speed. In this respect, CAVs typically take advantage of V2V and/or V2I communication environments. V2V communications can provide direct data transfer with a much lower delay compared to radars [21]. The V2V communications enable vehicles to drive closely with short inter-vehicular distances. This will increase the amount of road throughput and reduce the need for developing more road network. The vehicles can exchange data, such as inter-vehicular distance, speed and acceleration. In this context, Cooperative Adaptive Cruise Control (CACC) system has been widely developed featuring the possibility of coordination between connected vehicles aiming at enhancing the fuel efficiency, safety, driving comfort, and road throughput. This system, that is the advanced version of Adaptive Cruise Control (ACC), lets neighboring vehicles form a platoon which is a string of

Figure 1.2: Scania truck platooning [20]

vehicles following a common speed profile (see Fig. 1.2 for instance).

Vehicle platooning can exhibit various scenarios, such as single-lane platooning or multi-lane platooning. Many research works have tackled both of these problems. For instance, the works [22–24] have focused on the single-lane platooning and authors of [25–27] studied the multi-lane platooning problem. In a multi-lane platooning scenario, which is a generalization of single-lane platooning, there are at least two strings of CAVs driving with possible different speeds in adjacent lanes. In this situation some vehicles might decide to perform a cut-in or cut-out action to exit their own platoon and merge with the neighboring one safely. It is also notable that simpler cases of this problem, such as general lane change or on-ramp merging have already been studied [28, 29]. For instance, in on-ramp merging scenario a vehicle tries to safely merge into a platoon from a ramp (see Fig. 1.3) [29]. Some research works have also focused on studying interaction protocols for cooperative and highway platoon merging [30–32].

At the heart of developing AV and CAV, *control design* is one of the most prominent tasks which needs to be taken into account with significant care. Design of the intended controller tackles numerous challenges. One of the most significant classes of these challenges is the presence of unknown uncertainties which can unpredictably affect the resulting system. We will focus on these uncertainties in the current research. These uncertainties include a vast range of unknown situations, such as driving behavior, road friction variations, and so on [33, 34]; however, the ones that we are going to take into account include acceleration attacks and V2V/V2I communication attacks such as communication delays

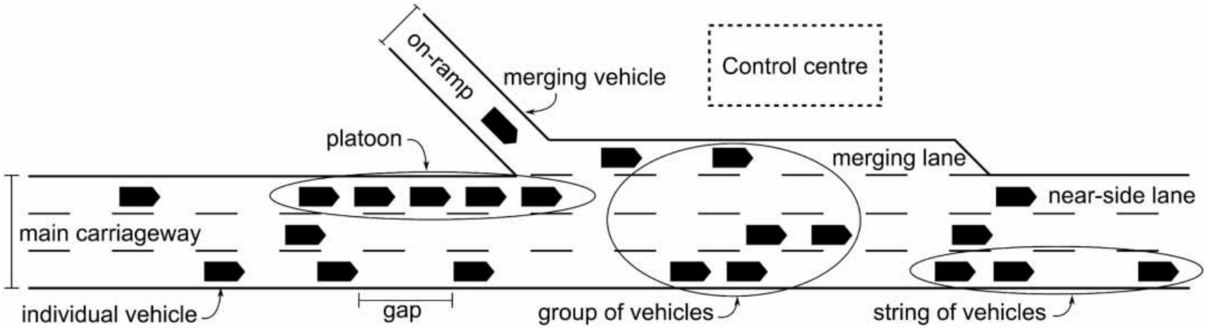


Figure 1.3: Single-lane platooning with an on-ramp merging formation [29]

(see Fig. 1.4).

1.2 Motivation

Existence of adversarial attackers makes the problem of controlling a vehicle platoon challenging. Due to the presence of such attackers, the normal operation of a platoon is surely endangered. This highlights the need for secure control methodologies for a safe platooning experience. As was explained before, CAVs forming platoons have significant environmental and transportational benefits. The simple case is the single-lane platoon in which a string of connected vehicles coordinate with each other to safely follow a common speed profile generated by the leader vehicle. The platoon is required to keep the minimum safe distance between the sequence of vehicles. In practice, it is most often the case where several platoons with possible different speeds trip along a multi-lane highway. In this case, the vehicles need to be ensured that they can safely and securely perform cut-in/cut-out maneuvers to travel between different adjacent platoons. *Homogeneous platoons* consist of identical vehicles with same model dynamics. However, existing of different vehicles with different model dynamics results in even a more challenging scenario called a *heterogeneous platoon*. In this scenario, it is important that regardless of variety in the model of the vehicles the control system can still ensure the safety of the platoon in addition to providing a safe environment for cut-in/cut-out actions. Besides, exposure of the communication among the vehicles to attacks may make the situation even worse.

Various control methods such as adaptive control, robust control, optimal control, Model Predictive Control (MPC), Stochastic-MPC (SMPC), have been largely utilized to tackle autonomous driving challenges in different scenarios like unsignalized intersection,

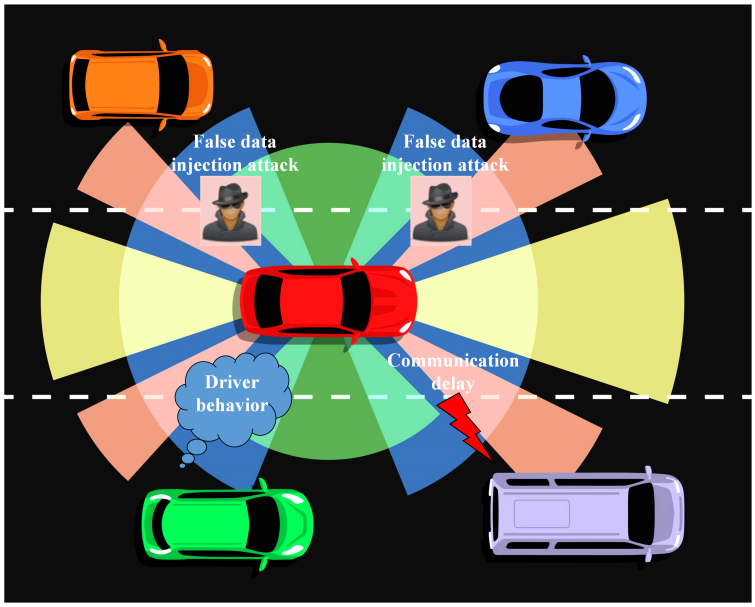


Figure 1.4: Possible existing uncertainties in an autonomous driving context

roundabouts, uncertain environments, and multi-lane traffic situation [35–40]. Although these methods have shown promising accomplishments in single autonomous driving context, when the connectivity notion arises in a set of connected vehicles, the aforementioned methods have significant limitations confining their applicability in these scenarios where connectivity issues such as cyber attacks and communication delays might occur.

With the above mentioned explanations in mind, it is highly significant to address the vehicle platooning problem along with its security related issues (in the presence of cyber attacks). This motivates us to develop control methods which then can be utilized in the context of CAVs in order to have a safe and secure connected autonomous driving experience.

1.3 Objectives

This thesis is structured into two main parts: I) High level secure platoon control, and II) Low level secure platoon control

High level secure platoon control: In this part, we consider platoons with arbitrary inter-vehicular communication topology whereby the vehicles are able to exchange

their driving data with each other through DSRC-based environment. The whole platoon is modeled using graph-theoretic notions by denoting the vehicles as the nodes and the inter-vehicular communication quality as the edge weights. Although we consider linear second/third-order models for vehicle dynamics, this is to define and fulfill the platoon control objectives while not bothering ourselves with control input signal such as the amount of torque sent to the wheels.

Low level secure platoon control: In this part, the platoon can still encompass arbitrary wireless connectivity among the cars; however, we pay more attention to the nonlinear dynamics of the participating vehicles in the platoon and will not linearize the model based on which new nonlinear control algorithms are introduced accordingly. In addition, we let the vehicles bear different dynamics to form nonlinear heterogeneous platoons. The required amount of control signals (such as torque) sent to the wheels to meet the control objectives and also the resulting absolute/relative position, speed, and acceleration of each of the vehicles are given.

The main objectives of this thesis are as follows:

- High level secure platoon control:

  1. The first objective is to study the security of a vehicle platoon equipped by unidirectional or bidirectional communication links among the vehicles using a game-theoretic approach. We aim at defining a game wherein the attacker and the detector are its players. The attacker targets some vehicles in the platoon to attack and the detector deploys monitoring sensors on the vehicles. The attacker's objective is to be as stealthy to the sensors as possible while the detector tries to place the monitoring sensors to detect the attack impact as much as it can (Chapter 3).
  2. The second objective is to find a novel optimal actuator placement strategy according to a defined Stackelberg game between the attacker and the defender. Based on the defined game and its optimal equilibrium point, the defender(s) selects optimal actuator placement action to face the attacker(s) (Chapter 4).

- Low level secure platoon control:

  3. The third objective is to tackle a common cyber attack, namely the Denial-of-Service (DoS) occurred in the communication links of a static vehicle platoon via a new secure model predictive control approach. The controller has to be able to fulfill the platoon speed tracking and minimum safe desired space requirements

along with facing the DoS attack. The vehicles participating in the platoon are to modeled by nonlinear dynamics and could be different resulting in a heterogeneous platoon (Chapter 5).

4. The fourth objective is to extend the results to the dynamic platooning case wherein cut-in/cut-out maneuvers can also be performed by the vehicles participating in the platoon. This requires the controller to handle a DoS attacked dynamic nonlinear heterogeneous vehicle platoon (Chapter 6).

1.4 Thesis Layout

Chapter 3 proposes a game-theoretic approach to investigate the security level of a platoon exposed to cyber attack. The considered platoon can exploit direct or undirected data transfer links among the cars. The equilibrium of the defined game determines the optimal policy which has to be used by the detector to deploy the monitoring sensors to increase the security level of the system. This chapter also studies the effect of adding (or removing) communication links between vehicles on the game value. The simulation and experimental results conducted on a vehicle platoon setup using Robotic Operating System (ROS) demonstrate the effectiveness of the performed analyses.

In Chapter 4, we propose a general approach to find an optimal actuator placement strategy according to the Stackelberg game between the attacker and the defender. The game payoff is the energy needed by the attacker to steer the consensus follower-leader dynamics of the system towards his desired direction. The attacker tries to minimize this energy while the defender attempts to maximize it. Thus, based on the defined game and its optimal equilibrium point, the defender(s) selects optimal actuator placement action to face the attacker(s). Both cases of single attacker–single defender and multiple attackers–multiple defenders cases are investigated. Furthermore, we study the effects of different information flow topologies, namely the unidirectional and bidirectional data transfer structures. Besides, the impacts of increasing the connectivity among the nodes on the security level of the platoon are presented. Simulation results for h–nearest neighbor platoon formations along with experimental results using the scaled cars governed by Robotic Operating System (ROS) verify the effectiveness of the method.

Chapter 5 proposes a secure distributed nonlinear model predictive control algorithm consisting of i) detection and ii) mitigation phases. The algorithm is capable of handling DoS attack performed on a platoon equipped by different communication topologies and at the same time it guarantees the desired formation control performance. Stability analysis of

the attacked platoon running the given algorithm is also presented. Simulation results on a sample heterogeneous attacked platoon exploiting two-predecessor follower communication environment demonstrates the effectiveness of the method.

Chapter 6 extends the method given in Chapter 5 which is also capable of providing safe and secure control of dynamic platoons in which arbitrary vehicles might perform cut-in and/or cut-out maneuvers. Convergence time and stability analysis of the system are also investigated in some cases. Furthermore, to handle DoS attacks modeled by an exceeding time delay in inter-vehicular data transmission, we propose the integration of an Unscented Kalman Filter (UKF) design within the controller resulting in a novel Secure–DNMPC–UKF co-design. This, in essence, estimates the delayed system states and feeds the predicted values to the Secure–DNMPC, which efficiently mitigates the attack effects. Simulation results demonstrate the fruitfulness of the proposed method.

Finally Chapter 7 concludes the thesis and summarizes the presented contributions. It also presents some possible open avenues for further research.

Chapter 2

Literature Review and Background

2.1 Basics of Vehicle Platooning

2.1.1 Background

Safe and secure driving experience is one of the most significant objectives in recently emerging intelligent transportation systems [41, 42]. Evolution of smart and autonomous vehicles has highlighted this concern much more than the past decades [43]. On the other hand, the possibility of featuring the connectivity and cooperation of vehicles has led to the emergence of strings of connected vehicles, namely *platoons*. Platoons have provided the opportunity to enhance the driving safety, ecological performance, road throughput, and comfort level [15, 26, 44–50]. Current standards for vehicular communications enable cars to exchange data, such as inter-vehicular distance, speed, and acceleration among each other through different communication environments, namely Vehicle-to-Vehicle (V2V), Vehicle-to-Infrastructure (V2I), Vehicle-to-Cloud (V2C), Vehicle-to-Broadband (V2B), and Vehicle-to-Roadside (V2R) [51, 52]. V2V communications can provide direct data transfer with a much lower delay compared to radars [21] and enable vehicles to drive closely with short inter-vehicular distances. This will increase the amount of road throughput and reduce the need for developing more road network. In this context, Cooperative Adaptive Cruise Control (CACC) has been widely developed featuring the possibility of coordination between connected vehicles which aims to enhance the fuel efficiency, safety, driving comfort, and road throughput [21, 28, 53–60]. Different types of controllers such as model predictive control [61], and decentralized overlapping control [62], have been introduced to take advantage of this connectivity among the vehicles to improve driving

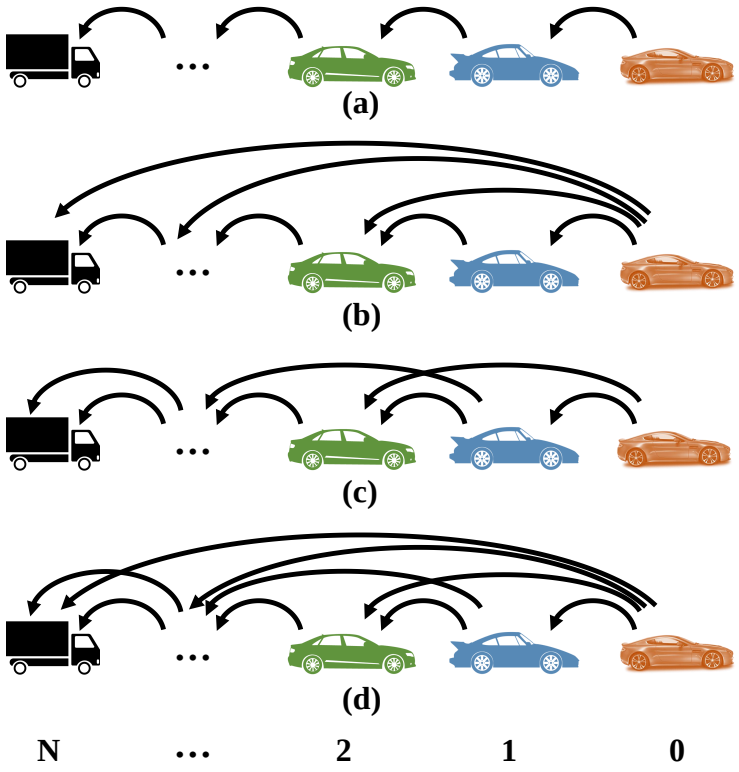


Figure 2.1: Unidirectional topology: (a) PF, (b) PLF, (c) TPF, and (d) TPLF, (vehicle 0 is the Leader Vehicle (LV))

experience [61,63,64]. Platoons could be formed based on different spacing policies and can be governed by different formation control techniques such as traditional linear/nonlinear controllers, optimal control methods, and more advanced consensus algorithms [65–67]. Getting more developed through using more effective data communication structures, connected vehicles are equipped with different information flow topologies to facilitate and improve the efficacy of data transfers. Predecessor-follower (PF), predecessor-leader follower (PLF), two-predecessors follower (TPF), two-predecessors-leader follower (TPLF), all-predecessors follower (APF), all-predecessors-leader follower (APLF), and h–nearest neighbor are some of the instances [68, 69]. These structures can be exploited either in a unidirectional or a bidirectional data transmit (see Fig. 2.1).

2.1.2 Vehicular Communication Standards and DSRC

In vehicle platooning, there have been some studies that investigated interaction protocols and standards for data sharing [30,31]. Other researches have also considered degradation and communication loss of data transfers affecting the CACC performance [70,71]. Generally, several important variants of wireless data transfer systems exploited in connected vehicles include DSRC, VANET, and MANET [72–74]. DSRC, which was developed by the American Society for Testing and Materials (ASTM), has been leveraged as one of the communication methods for V2X communications as an inter-vehicular communication infrastructure, and is largely based on IEEE 802.11p, which uses Carrier-Sense Multiple Access with Collision Avoidance (CSMA/CA) protocol. It was initially developed to operate on a 912 MHz bandwidth channel in 1992. In 2002 and 2003, it was improved as the particular standard used in Intelligent Transportation Systems (ITS) using IEEE 802.11a operating on the 5.9 GHz band denoted by E2203-02 and E2203-03, respectively [75]. Recently, this protocol, which is also called the Wireless Access in the Vehicular Environment (WAVE), is established to manage the data transfers in the 5.9 GHz band on seven different channels of 75 MHz bandwidth [76,77]. Each channel is 10 MHz wide along with 5 MHz reserved before the channels. For more details of DSRC and other protocols, the reader is referred to [76,77] and references therein.

2.2 Cyber Attacks on Vehicle Platoons

Despite plenty of benefits resulting from the use of wireless communications in a platoon, it is naturally vulnerable to cyber attacks. Different types of attacks on a platoon can be generally classified into three classes, namely application layer attacks, network layer attacks, and privacy leakage attacks [78]. All these attacks can potentially endanger the string stability of the platoon. Moreover, the attacker could be an external or an internal malicious agent performing each of the above-mentioned attacks [79]. For details of the aforementioned attacks, the reader is referred to [78]. False data injection attack (message falsification/tampering), replay attack, zero dynamics attack, covert attack, jamming attack, eavesdropping attack, Man-in-the-Middle attack, GPS spoofing, impersonation attack, masquerading attack, and Denial-of-Service (DoS) are some of the possible real-world attacks on vehicle platoons [80–83]. We will focus on bias injection attack (Chapters 3 and 4), and DoS attack (Chapters 5 and 6) as common forms of disruption attacks [84,85]. Another attack classification in literature splits the attacks into control algorithm modification and sensor reading tampering classes [79]. Control algorithm modification attacks

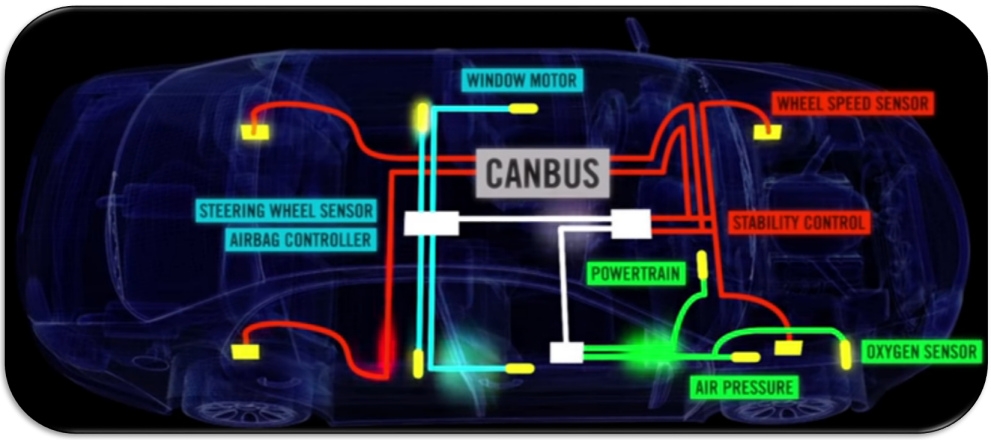


Figure 2.2: CAN bus connecting different internal vehicle components [91]

include destabilizing attacks [86], high-speed collision induction attacks [87], and traffic flow instability attacks [88, 89]. Sensor reading tampering attacks consist of false data injection [90] and efficiency-motivated attacks [82].

It is notable that the vulnerability of a platoon against attacks can also arise from insecure individual vehicles participating in the platoon. Therefore, securing single vehicles individually is also essential to ensure the security of the connected vehicles in a platoon. In this respect, a means of attacks on a single vehicle is to exploit the vulnerability of a component of the vehicle allowing access to its Controller Area Network (CAN) bus. For instance, in several real car attacks occurred recently, the infotainment component of the vehicle that required no authentication and could be accessed anonymously, was exploited aiming at getting access to the CAN bus of the vehicle thereby attaining control on different operations of the car, such as steering wheel, engine, and braking system (see Fig. 2.2). For a comprehensive list of real-world attacks happened on vehicles from 2010 up to 2017 together with several countermeasures, the reader is referred to [92]. In the sequel, we take a look at the practical considerations and a more detailed classification of cyber attacks imposed on single and connected vehicles.

2.2.1 Practical Considerations

In this section, we will go through further in the details of practical aspects of possible cyber attacks on vehicle platoons. We discuss different attack classifications and explain the practical issues that might occur while the attacks are performing.

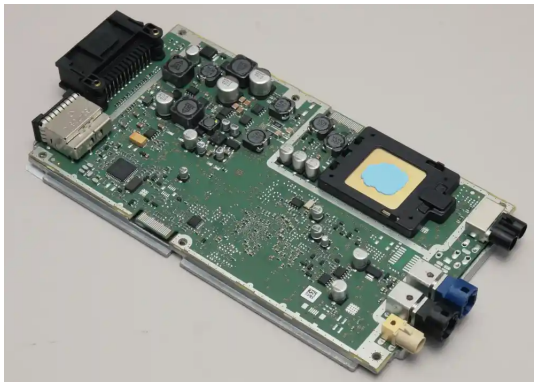


Figure 2.3: Insecure Combox used in BMW attack [94]

Under-the-Hood Elements

From a practical point of view, there are several different components in a vehicle which can be under the risk of attacks. Under-the-hood elements, such as ECUs, vehicular network, and the communication gateway are some of these components [93]. An attacker can exploit possible existing backdoors of an ECU. Running operating systems (to handle high burden functionalities), ECUs cannot close all the intrusion vulnerabilities of a typical operating system. Besides, the manufacturers have to leave some open tunnels in ECUs for diagnostic purposes. These limitations can be abused by an adversary for devastation objectives.

Vehicular Networks

Vehicular networks are other components which need to pay sufficient care to during and after manufacture. Typically, these networks are designed such that they remain isolated from an outer world like the Internet and hence developed mostly to operate in closed networks. Thus, in principle, they lack security protections and are proper backdoors for the attacker to penetrate the internal vehicle operations.

Gateways

Gateways are the connection bridges between vehicular networks and standard communication protocols, such as Bluetooth or WiFi. BMW ConnectedDrive attack is a real-world cyber attack suffered from an insecure employed gateway in the vehicular networks [94].

Mainly, the adversary could exploit the vulnerability in the modem of the Combox shown in Fig. 2.3 to unlock the car. It is noteworthy that the Combox was mainly designed as an infotainment unit which is also responsible for the handling of emergency calls. Generally, in case the gateways use their own operating systems, a spoofing attacker can replace the operating system with his own software. Moreover, as the gateways are the point where communication protocols and vehicular networks meet, a Denial-of-Service or Man-in-the-Middle attack[1] can break this connectivity resulting in performance degradation. For the details regarding the vulnerabilities of mobile devices and communications used in vehicles, the reader is referred to [93] and references therein.

As was mentioned before different types of attacks on a platoon can be generally classified in terms of the level of the platoon architecture on which the attack is imposed or based on the security issue they raise [78, 79]. In the following, we explain different attack classifications in detail. Fig. 2.4 illustrates different possible attacks in connection with different internal vehicle parts.

2.2.2 Application-Layer, Network-Layer, System-Level and Privacy Leakage Attacks

In this section, we study different types of attacks that might occur in vehicle platoons in practice. We present a classification of attacks in terms of the level of the platoon architecture on which the attack is imposed. This classification subdivides the attacks into four main classes, namely application layer attacks, network-layer attacks, system-level attacks, and privacy leakage attacks [78].

Application Layer Attacks

Several important variants of wireless data transfer systems exploited in connected vehicles include DSRC, VANET, and MANET [72, 73]. DSRC has been leveraged as a standard protocol for inter-vehicular communication infrastructures.[2] This protocol is established to manage the data transfers in the 5.9 GHz band on a 75 MHz bandwidth channel [77, 95]. For the details of other protocols, the reader is referred to [51, 52] and references therein. Application layer attacks can endanger the functionality of the aforementioned protocols and other applications, such as CACC message beaconing in the data sharing protocols. Message falsification, GPS spoofing, and replay attacks lie under this class.

[1] Details of such attacks will be explained subsequently.

[2] There have been some studies that investigated interaction protocols for data sharing [30, 31].

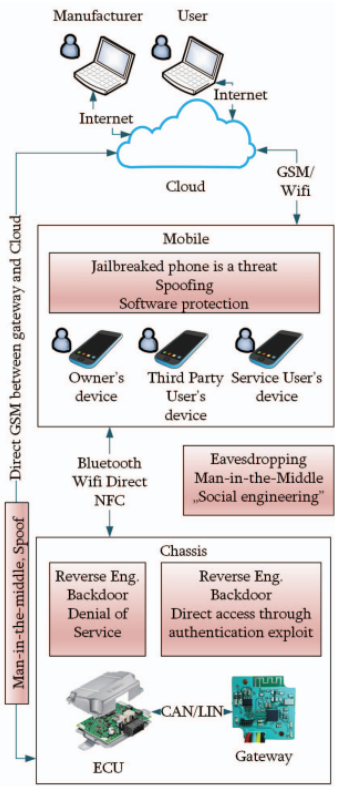


Figure 2.4: Different internal vehicle components and possible attacks [93]

Application layer attacks can surely degrade the string stability of the platoon and might cause accidents.

In message falsification attack, the attacker starts listening to a beacon in the wireless medium and then rebroadcasts the falsified version of the received messages. For instance, the adversary can manipulate the received acceleration of a preceding vehicle and transfer the modified value to the following cars.

Spoofing attack relates to impersonation of the attacker aiming at injecting fraudulent data to the other vehicles. As an example, consider a predecessor-follower communication topology among the platoon. Having performed the spoofing mechanism, the adversary impersonates himself as the preceding vehicle of the target one and starts sharing incorrect data to the follower. This, in turn, causes the propagation of false data through the platoon resulting in hazardous actions or even accidents.

In a replay attack, the attacker receives and stores the data and replays it at a later

time. The replayed data contains old values which are no longer valid in the new platoon situation. Assume that the platoon speed was already 100 km/h which is received and stored by the attacker. When the leader slows down to 70 km/h, the attacker replays the previously stored value of 100 km/h to the followers causing hazardous collisions.

Network Layer Attacks

A network layer attack is mainly related to DoS or Distributed DoS (DDoS) attacks. In these attacks, the communication capability of the vehicles in the platoon is subject to overwhelm. Particularly, in the existing network between the vehicles there exists a specified component called Hardware Security Module (HSM) which is responsible for storing digital keys as well as performing all cryptographic operations, such as message signing/verification, encryption, and hashing. The HSM can handle a limited number of the aforementioned tasks at a time as they are computationally complex and require massive computational power. Hence, the DoS attack can target this limitation to overwhelm the communication capability of the platoon.

Radio jamming is another DoS attack that aims communication protocols such as commonly used IEEE 802.11p standard. Specifically, this standard uses one control channel (CCH) with multiple service channels (SCHs). The adversary can use different jamming techniques like one-channel jamming or swiping between all channels and trying to jam them all.

System Level Attacks

The previous attacks are all to manipulate the data transfer mechanism in the wireless infrastructures. On the other hand, the system level attacks, are related to manipulation of hardware/software of the vehicles. This can be generally done by an insider at the manufacturing level or by an outsider in an unattended vehicle (e.g., by replacing or altering certain vehicle sensors). The significance of this kind of attacks is apparent, i.e., even if the V2V communications among the vehicles are error-free and completely secure, the on-board hardware/software of the vehicles can still be tampered causing performance degradation.

Privacy Leakage Attacks

Through the transferred data between the vehicles, there might be private data containing valuable information about the platoon formations. These data can be the platoon length,

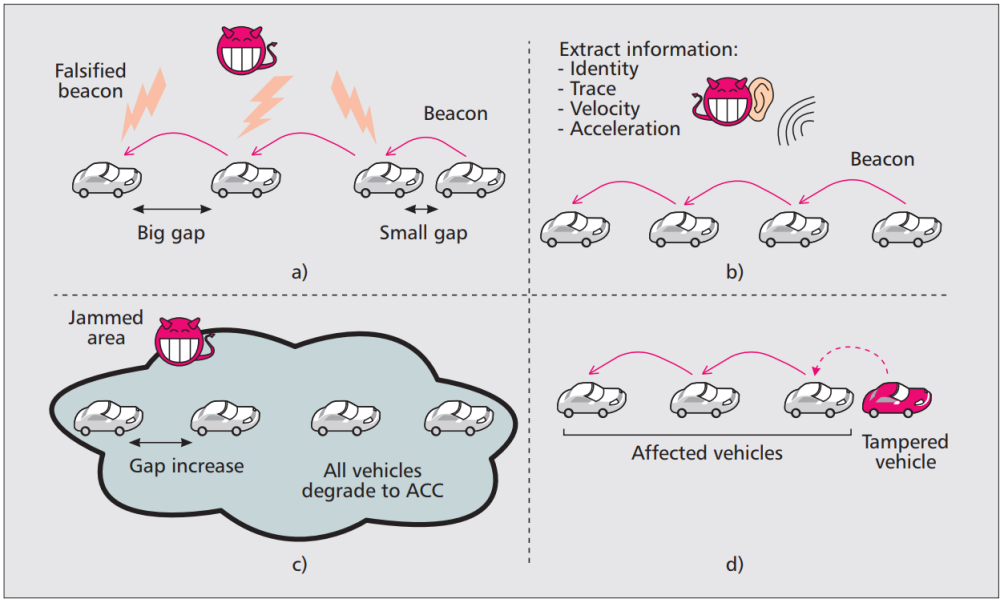


Figure 2.5: Different attack types in a vehicle platoon: a) Falsification attack; b) Eavesdropping attack; c) Radio jamming attack; d) Tampering attack [78]

the vehicles' identity participating in the formation, and the intended speed/acceleration of the platoon. Consequently, preserving data privacy is another concern which can be threatened by malicious agents. Privacy leakage attacks can even happen easier in case the data exchanged among the platoon are signed allowing the adversary to easily identify the participating vehicles in the CACC stream.

Eavesdropping is a passive attack abusing the above-mentioned limitations and vulnerabilities. These attacks can be avoided by encryption and/or anonymity techniques using group signatures or short-term certificates (pseudonyms) [96, 97]. Fig. 2.5 illustrates the above-mentioned attacks schematically.

2.2.3 Data-Related vs. Control-Based Attacks

Another interesting attack classification is based on the security issue they raise [79]. From this perspective, the attacks can be split into two main categories. The first category consists of vehicular network attacks which are related to data threatening. They can be categorized into four major classes, namely the attacks that cause missing data availability, those causing missing data confidentiality, attacks aiming at missing data authentication,

and attacks resulting in losing data integrity. The second category comprises the attacks that endanger the platoon control systems. This category is comprised of attacks whether they put control algorithms or sensor readings under risk.

Data-Related Attacks

Data-related attacks largely threaten the vehicular network of the platoon. In particular, they can be classified as follows,

1. Missing data availability
    - Jamming attack
    - DoS
2. Missing data confidentiality
    - Eavesdropping attack
    - Man-in-the-Middle attack
3. Missing data authentication
    - GPS spoofing
    - Impersonation attack
    - Masquerading attack
    - Message tampering
4. Losing data integrity
    - Replay attack
    - Message modification attack

Control-Based Attacks

Control-based attacks mainly endanger the platoon control systems. They can be classified as follows,

1. Control algorithm modification

- Destabilizing attack
- High-speed collision induction attack
- Traffic flow instability attack

2. Sensor reading tampering
    - False data injection attack
    - Efficiency-motivated attack

2.3 Secure Vehicle Platooning

Back to the platoons, an adversarial attacker can target one or several vehicles to physically/remotely manipulate the sensors of the vehicles which can eventually cause hazardous actions or even accidents. This signifies the importance of security of the vehicle platoons against external malicious attacks. Hence, the need for monitoring systems capable of detecting the attackers' action is highly sensible [98–104]. One of the important aspects of deployed monitoring sensors is their location regarding the possible locations of injected attacks. Consequently, it is largely essential to have a systematic procedure based on which the detector can place its sensors on specific locations to increase the security level of the system.

Recently, much research has been done in investigating the security of networked control systems from various perspectives [105–109]. Communication-related protection methods, such as encryption of wireless channels, are techniques to avoid receiving compromised data via the wireless infrastructures [52]. On the other hand, control-oriented concepts, such as game-theoretic methods are also among the leading methodologies which address the security issue of general cyber-physical systems with a considerable amount of care [110, 111]. Various approaches, such as Nash or Stackelberg formulations, demonstrate the conflicting decisions between the players (attackers and defenders) [112,113]. The existence of an equilibrium state for this game is a solution based on which the detector can decide about its sensor placement strategy. Cooperative games are some other recent approaches aiming at modeling networked control systems [114]. Based on these games, robustness analysis of the system against malicious attacks has also been studied [115–117]. With respect to securing communication protocols used in platoons, secure communication protocols for VANETs based on game theory have been proposed either for multimedia transmission [118] or for communications exposed to specific attacks [119]. Network-aware control methods have also been proposed to handle possible communication failures through the platoon. Those

approaches mainly consider random communication failures with an emphasis on the control/stability performance of the whole platoon without considering intelligent cyber attacks [120–122]. Despite the above-mentioned works, a general procedure for investigation of security of vehicle platoons under cyber attacks in which the quality of communications among the vehicles are different is missing and has not been addressed yet. Game-theoretic approaches provide a powerful tool to tackle the attacker-detector conflicting actions as an attacker-detector game and study the security of a platoon based on various decisions made by the adversarial and the defender. Hence, one of the contribution of the current thesis is to formulate the security problem of a general platoon, which is under cyber attacks, as a game where both the attacker and the defender attempt to face each other in opposite ways. Moreover, the communication links between different vehicles can have different qualities and both the unidirectional and bidirectional data transfer structures are taken into account in this work. More rigorously, the adversary tries to attack specific vehicles of the platoon such that he remains undetected while the defender endeavors to locate his sensors on specific vehicles such that the detectability of the attacker is maximized, hence, increases the security level of the system.

As was mentioned before, vehicle platoons lie under a wide class of newly emerged systems, namely Cyber-Physical Systems (CPS). When CPS comes into view, control, computation, and networking bind together to form a suitable infrastructure for control and systems purposes. Having taken advantage of networking and wireless communications, CPS could develop a vast range of large scale widespread control systems. However, this might bring up a substantial challenge, which is the vulnerability of CPS against cyber attacks [7]. Although there has been a large amount of research addressing the security of CPS, those systems still suffer from the lack of secure performance in the presence of possible malicious intruders [109, 123, 124]. This should also be noted that although some techniques introduced in the fault-tolerant control are applicable to security problems, most of them have been shown to fail to mitigate the effects of an attack [125]. This is due to the fact that an attacker is intrinsically an intelligent agent who might have some *a priori* knowledge about the system dynamics and the controller contrary to a random fault. Furthermore, an intelligent adversary might target specific components of a system based on his own criteria, such as optimizing the amount of consumed energy or the intended level of devastation. In this respect, various system-theoretic, graph-theoretic, and game-theoretic approaches [111, 126–128] have been proposed in the last decade to address security issues of general control systems. Vehicle platoons, as a large class of CPS, incorporate the physical dynamical systems, referred to as the physical layer, along with the wireless communication systems indicating the cyber layer [15]. Therefore, these systems need to guarantee a safe and secure performance in the case of dealing with unreliable and

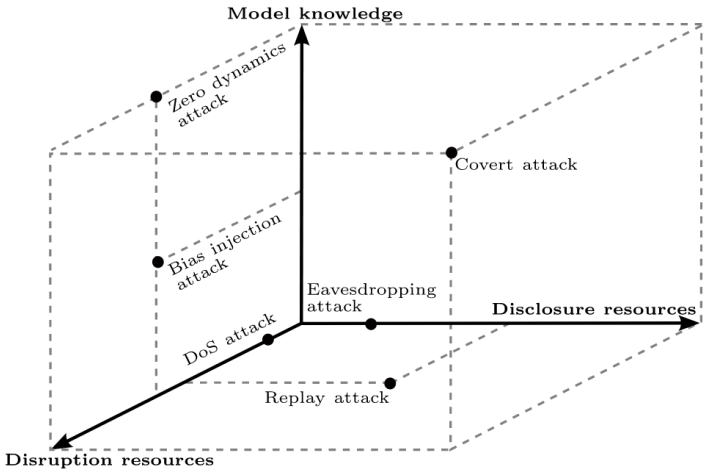


Figure 2.6: Cyber attack classification in a three-dimensions attack space [84]

compromised networks. In recent years, researchers have been concerned about possible vulnerabilities of vehicle platoons against cyber attacks as well as communication delays [46,70,71,83,86,122,129–135]. For instance, in [129], a DoS attack detection and estimation scheme based on sliding mode observer has been proposed for a linear homogeneous car following system. Also, authors of [70, 71] study the performance degradation of a linear homogeneous vehicle following controller caused by unreliable wireless communications. Various types of intrusions imposed by either insider or outsider adversary on connected vehicles such as GPS spoofing, DoS, masquerading, insider/outsider eavesdropping have been investigated in literature [136]. Each of these attacks can degrade system performance by violating one or more of the data integrity, data availability, and data confidentiality (see Fig. 2.6). A detailed and formal attack classification in a three-dimensions attack space is presented in [84]. Hence, one of the most prominent aspects of vehicle platooning is to ensure its security while one (or more) intruder(s) intend to devastate the performance of the platoon by injecting attack signals to one or more components of the system [93]. The injected attack signal could be of a physical one affecting the dynamical quantities of the vehicles or any deterioration of the communication framework existing among the nodes. Adversarial attackers can even access the control of the vehicles remotely, hence, largely endanger the safety of the platoon. For instance, false data injected by a replay intruder, which seems admissible to the system and the controller, might cause violation of the safe desired gap among consecutive vehicles and result in severe accidents [78]. As another example, in GPS spoofing, legitimate GPS signals are interfered with by the attacker who transmits inaccurate coordinates. This will fool the controller of an ego-

vehicle; thus, incorrect torque input is applied to the wheels, which might cause collisions. The vulnerability of a platoon against attacks can also be caused by insecure individual vehicles participating in the platoon. Thus, securing single vehicles individually is also essential to ensure the security of the connected vehicles as a whole. In this regard, a possible means of attack on a single vehicle could be to exploit the vulnerability of one of the components of the vehicle allowing access to its CAN bus [137]. This has been the case in several real car attacks occurred recently [92]. For instance, the attacker might access remotely to the CAN bus of the car through unauthorized infotainment facility, thereby take the full control of acceleration or braking operations [138].

Over the recent years, researchers have introduced algorithms to overcome the security threats of connected vehicles with different approaches [78]. For instance, [139] views the problem by capturing the control-theoretic methods to address the resiliency of the connected vehicles against the adversary. Encryption of the data transmitted through the platoon, Quality of Service (QoS), and safety awareness of the vehicles are other techniques that have been exploited to address the privacy leakage of a platoon [41, 80, 140, 141]. Furthermore, safety issues that may arise due to possible malfunctioning of some redundant sensing/communication devices installed on the vehicles have been addressed in literature [142, 143]. Observer-based techniques have also been introduced to tackle the packet drop phenomenon in the network among the vehicles without compromising the overall performance [129]. Besides, game-theoretic approaches, in some instances interlaced with graph-theoretic techniques, have been shown in several paradigms to be quite fruitful to confront the security issues of control systems [111, 114, 132, 144–146]. For instance, authors in [111], employed a game-theoretic framework to confront the jamming attack threatening remote state estimation in general CPS. In [146] the minimum relative distance among two consecutive vehicles is derived based on a game-theoretical optimal control scheme. The authors have verified that this minimum safe distance heavily depends on the maximum deceleration ability of the follower vehicle. In [145], the authors study the optimal control action for a standard discrete-time linear quadratic Gaussian system in the presence of an intelligent intruder, who jams the communication link among the controller and the plant, by defining Nash/Stackelberg game problems. It is worth mentioning that Stackelberg game formulation is more applicable to most of the security problems [147]. This is due to the fact that in most of these problems the leader (usually the defender) designs his strategy based on a possible worst case attack scenario that can be reasonably captured by the Stackelberg game. Besides, these games always admit an equilibrium point which determines the optimal strategy for the defender, hence, guarantees the existence of a solution for the defender. In [114], the evolution of the networked control system is cast as a consensus model within the cooperative games in order to apply the potential games to

cooperative control problems. Those cooperative games were also investigated in terms of their resiliency against the communication failures imposed by an intruder [148, 149]. As such, vehicle platooning equipped by inter-vehicular data connectivities and formation controllers could inherently benefit from the mentioned literature to address the security challenges during attacks performed by an external adversarial attacker.

To date, there is no *systematic* and *general* procedure to mitigate the intelligent attack effects imposed on a platoon with arbitrary internal communication topology and formations. In fact, the current literature suffers from the lack of a general work-around for secure platooning which is independent of the employed communication topology, number of attackers, and attack location. This is what we are going to address in the current thesis from both the high level and low level perspectives.

Part I

High Level Safe and Secure Vehicle Platoon Control

Chapter 3

Security-Aware Optimal Sensor Placement in Vehicle Platooning

In this chapter, we study the security of a vehicle platoon exposed to cyber attacks using a game-theoretic approach. The platoon topologies under investigation are directed (called predecessor following) or undirected (bidirectional) weighted graphs. The edge weights specify the quality of the communication links between the vehicles in both the unidirectional/bidirectional data transfer environments. The attacker-detector game is defined as follows. The attacker targets some vehicles in the platoon to attack and the detector deploys monitoring sensors on the vehicles. The attacker's objective is to be as stealthy to the sensors as possible while the detector tries to place the monitoring sensors to detect the attack impact as much as it can. The existence of Nash Equilibrium (NE) strategies for this game is investigated based on which the detector can choose specific vehicles to put his sensors on and increase the security level of the system. We benefit from the system L_2-gain from the attack signal to the sensor measurements vector to characterize the cost function introduced in the game and determine the optimal sensor placement strategy. Moreover, we study the effect of adding (or removing) communication links between vehicles on the game value. The simulation and experimental results conducted on a vehicle platoon setup using Robotic Operating System (ROS) demonstrate the effectiveness of our analyses. Explicitly, our contributions in this chapter are as follows,

- For both predecessor-following (directed) platoon and symmetric (undirected) platoon, we investigate the existence of a Nash Equilibrium (NE) strategy for an attacker-detector game based on which the detector can place its sensors on specific nodes increasing the security level of the system. We consider the case of single attacked

vehicle, $f = 1$, as well as multiple attacked vehicles, $f > 1$, where f is the number of attacked nodes and deployed sensors on the network (Theorem 3.8, Theorem 3.10, Theorem 3.11, Theorem 3.12).

- We study the effects of adding or removing communication links (or weights) to (or from) the platoon on the game pay-off. Both undirected and directed scenarios will be investigated in this study, and we show that the behaviour of the game value in response to such topology variations is different for directed and undirected networks (Theorem 3.14, Theorem 3.15).
- The security level of a platoon equipped with undirected communication links among its vehicles will be compared to that of a platoon equipped with directed communication links. Our results show that using undirected data transfer increases the security level of the system which is consistent with the fact that the two-way data transfer between the pairs of vehicles lets them receive the attack signal from multiple ways instead of a single path, hence, resulting in a more reliable platoon (Proposition 3.17).

The results of this chapter have been published in [132].

3.1 Organization of the Chapter

This chapter is organized as follows. Sec. 3.2 defines the problem formulation of a platoon under cyber attacks. The attacker-detector game is defined and the system and attack modeling are presented in this section. In Sec. 3.3 we perform the equilibrium analysis for both the weighted undirected and directed data transfer scenarios where the attacker attacks one node and the detector places one sensor on a node. Sec. 3.4 extends the results to the case where more than one nodes are attacked and more than one monitoring sensors are supposed to be deployed. In Sec. 3.5 effects of adding extra communication links to a platoon are studied. Security level of a platoon with bidirectional versus unidirectional communication links is investigated in Sec. 3.6. Sec. 3.7 presents the simulation and experimental results. Finally, Sec. 3.8 concludes the chapter.

3.2 Problem Formulation

First of all, we present the notations and definitions used in the rest of the chapter for the sake of legibility.

3.2.1 Notations and Definitions

We denote a weighted undirected graph by $\mathcal{G}_u(\mathcal{V}, \mathcal{W})$ where $\mathcal{V}$ is the set of nodes (vertices) and $\mathcal{W}$ is the set of undirected edge weights. Assume $|\mathcal{V}| = n$. We note that $w_{ij} \geq 0$ for all $i, j = 1, 2, \ldots, n$ and $w_{ii} = 0$ for all $i = 1, 2, \ldots, n$. We say that (v_i, v_j) is an edge if and only if $w_{ij} > 0$. The leader node in a path graph is denoted by v_ℓ. For simplicity we define the weight of edge (v_i, v_j) by w_j if v_i is closer to the leader node. We denote a weighted directed graph by $\mathcal{G}_d(\mathcal{V}, \mathcal{W})$. We assume only unidirectional edges for the directed graphs, i.e., if there exists a directed edge from v_i to v_j in $\mathcal{G}_d$, then there is no directed edge from v_j to v_i. The adjacency matrix of $\mathcal{G}_d$ is $A_{n\times n}$ where $A_{ij} = w_i$ if and only if there is an edge from v_j to v_i. The *neighbor* nodes of vertex $v_i \in \mathcal{V}$ in $\mathcal{G}_d$ are determined by the set $\mathcal{N}_i = \{v_j \in \mathcal{V} \mid (v_j, v_i) \in \mathcal{G}_d\}$. The in-degree of node v_i (degree for undirected networks) is determined by $d_i = \Sigma_{v_j \in \mathcal{N}_i} A_{ij}$. The Laplacian matrix of a general graph $\mathcal{G}$ is defined as $L = D - A$, where $D = \text{diag}(d_1, \ldots, d_n)$. It is noteworthy that since we consider a general weighted graph, the degree matrix does not measure the number of outgoing and incoming edges, hence, does not only take values from the natural domain. In this chapter, we denote a vector which has a one in the i^{th} position and zero elsewhere by $\mathbf{e}_i$.

3.2.2 System Modeling

Let us consider a string of n connected vehicles in a platoon modeled by a weighted path graph $\mathcal{G}(\mathcal{V}, \mathcal{W})$. The edge weights are to model the communication quality between the vehicles. In practice, different scenarios could occur affecting the quality of data transfer between the vehicles.[1] It is notable that in DSRC-based communications, it is common to normalize the communication perfection of signals versus the sent power or distance. Hence, from a practical point of view, the edge weights used in this chapter can be normalized based on the above concepts to let the weight values lie in the $[0, 1]$ range; however, this is out of the scope of this chapter. Let p_i denote the position of vehicle v_i. The objective is for each vehicle to maintain a specific distance from its neighbors. The desired vehicle formation will be formed by a specific constant distance Δ_{ij} between vehicles v_i and v_j, which should satisfy $\Delta_{ij} = \Delta_{ik} + \Delta_{kj}$ for every triple $\{v_i, v_k, v_j\} \subset \mathcal{V}$. Considering the fact that each vehicle v_i has access to its own position, the positions of its neighbors, and the desired inter-vehicular distances Δ_{ij}, the control law for vehicle v_i is [154]

$$\ddot{p}_i(t) = \sum_{j \in \mathcal{N}_i} k_p \left(p_j(t) - p_i(t) + \Delta_{ij}\right) + k_v \left(\dot{p}_j(t) - \dot{p}_i(t)\right) + \zeta_i(t), \tag{3.1}$$

[1] For instance, entering the platoon in a long tunnel may degrade the proper data transfer among the vehicles [150–153].

where $k_p, k_v > 0$ are control gains and $\zeta_i(t)$ models the injected attacks. Physically, this means that the attacker adds a traction acceleration (or brake) to vehicle v_i. Dynamics (3.1) in matrix form become

$$\begin{cases} \dot{\boldsymbol{x}}(t) = \begin{bmatrix} \mathbf{0}_n & I_n \\ -k_p L_g & -k_v L_g \end{bmatrix} \boldsymbol{x}(t) + \begin{bmatrix} \mathbf{0}_{n\times 1} \\ k_p \boldsymbol{\Delta} \end{bmatrix} + \begin{bmatrix} \mathbf{0}_n \\ B \end{bmatrix} \boldsymbol{\zeta}(t), \\ \boldsymbol{y}(t) = \begin{bmatrix} C & 0 \end{bmatrix} \boldsymbol{x}(t), \end{cases} \tag{3.2}$$

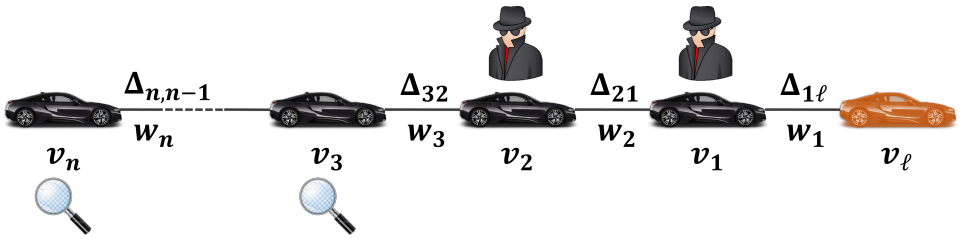
where $\boldsymbol{x} = [\mathbf{P} \quad \dot{\mathbf{P}}]^\mathsf{T} = [p_1, p_2, \ldots, p_n, \dot{p}_1, \dot{p}_2, \ldots, \dot{p}_n]^\mathsf{T}$, $\boldsymbol{\Delta} = [\Delta_1, \Delta_2, \ldots, \Delta_n]^\mathsf{T}$ in which $\Delta_i = \sum_{j\in\mathcal{N}_i} \Delta_{ij}$. Here L_g is the grounded Laplacian matrix which is the reduced Laplacian matrix by removing the row and the column corresponding to the leader node, $\boldsymbol{y}(t)$ is the sensor measurements vector, and $\boldsymbol{\zeta}(t)$ is the attack vector. Matrices B and C represent the attacker and detector decisions, respectively. For instance, let us consider a specific vehicle platoon with $n = 4$ vehicles subject to cyber attacks shown in Fig. 3.1. Suppose that the attacker targets vehicles v_1 and v_2 while the detector places its sensors on vehicles v_3 and v_4. This gives $B = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}^\mathsf{T}$ and $C = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$. The reason that the positions of vehicles are our output of interest is that we need the vehicles' positions to guarantee the desired inter-vehicular distance in terms of safety of the platoon. These data are available through both GPS and on-board sensors of the vehicles. In order to prevent a possible misconception that might arise due to the usage of the word "sensor" for the defender's action, we state that this is the exact same means to measure the output of the system. Based on (3.6), the output of the system is the relative position of the vehicles. To measure the position of the ego-vehicle, this quantity can be measured by the sensors mounted on it such as a GPS. To measure the relative position of the other vehicles, different commonly used sensors can be utilized such as radar and Light Detection and Ranging (LIDAR) sensor. An example of undirected and directed platoons of n vehicles subject to two attacks and equipped with two monitoring detectors are shown in Fig. 3.1.

For the rest of our analysis, we derive a model for the error dynamics of the system (3.2). Let us denote the desired position of vehicle v_i in steady state by $p_i^*(t)$ and define the following tracking error

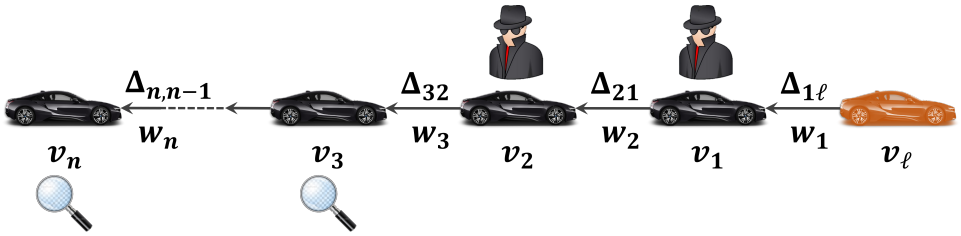
$$\tilde{p}_i(t) \triangleq p_i(t) - p_i^*(t). \tag{3.3}$$

Obviously, the desired formation of the platoon has to satisfy $p_i^*(t) = p_j^*(t) + \Delta_{ij}$ [155]. Substituting (3.3) in (3.1) yields

$$\begin{aligned} \ddot{\tilde{p}}_i(t) + \ddot{p}_i^*(t) = &\sum_{j\in\mathcal{N}_i} k_p \left(\tilde{p}_j(t) + p_j^*(t) - \tilde{p}_i(t) - p_i^*(t) + \Delta_{ij}\right) \\ &+ k_v \left(\dot{\tilde{p}}_j(t) + \dot{p}_j^*(t) - \dot{\tilde{p}}_i(t) - \dot{p}_i^*(t)\right) + \zeta_i(t), \end{aligned} \tag{3.4}$$



(a) Weighted undirected platoon with n vehicles



(b) Weighted directed platoon with n vehicles

Figure 3.1: Undirected and directed platoons with n vehicles and sample attackers and monitoring sensors

Now respecting the fact that in the steady state formation the vehicles' velocities have to be equal, we observe that $\dot{p}_i^*(t) - \dot{p}_j^*(t) = 0$. Furthermore, in the steady state formation the vehicles' velocities reach constant values which results in $\ddot{p}_i^*(t) = 0$. Hence, (3.4) is reduced to the following error dynamics model

$$\ddot{\tilde{p}}_i(t) = \sum_{j \in \mathcal{N}_i} k_p(\tilde{p}_j(t) - \tilde{p}_i(t)) + k_v\left(\dot{\tilde{p}}_j(t) - \dot{\tilde{p}}_i(t)\right) + \zeta_i(t), \tag{3.5}$$

The above error dynamics can be written in the matrix form as follows

$$\begin{cases} \dot{\tilde{\boldsymbol{x}}}(t) = \begin{bmatrix} \mathbf{0}_n & I_n \\ -k_p L_g & -k_v L_g \end{bmatrix} \tilde{\boldsymbol{x}}(t) + \begin{bmatrix} \mathbf{0}_n \\ B \end{bmatrix} \boldsymbol{\zeta}(t), \\ \tilde{\boldsymbol{y}}(t) = \begin{bmatrix} C & 0 \end{bmatrix} \tilde{\boldsymbol{x}}(t), \end{cases} \tag{3.6}$$

where $\tilde{\boldsymbol{x}} = [\tilde{\mathbf{P}} \quad \dot{\tilde{\mathbf{P}}}]^\mathsf{T} = [\tilde{p}_1, \tilde{p}_2, \ldots, \tilde{p}_n, \dot{\tilde{p}}_1, \dot{\tilde{p}}_2, \ldots, \dot{\tilde{p}}_n]^\mathsf{T}$ and the other variables are the same as in (3.2).

3.2.3 Attack Modeling: Bias Injection Attacks

For our particular application under study, i.e., vehicle platooning, we assume that the attacker does not inject a high frequency signal to the system. In fact, due to the large inertia of the vehicles, the attacker can not change the vehicle's acceleration abruptly. Hence, a high frequency attack signal which targets at changing the vehicle's acceleration can be immediately detected through receiving the information from the surrounding vehicles. Based on this fact, we consider a slowly time varying attack signal, namely a *bias injection attack*. Consequently, the L_2-gain of the system which equals the $\mathcal{H}_\infty$-norm of the system [156] can be calculated at the zero frequency.

Based on (3.6), the following proposition, formulates the system L_2-gain from the attack vector $\boldsymbol{\zeta}(t)$ to the output measurements vector $\tilde{\boldsymbol{y}}(t)$.

Proposition 3.1. *The system L_2-gain from the attack vector $\boldsymbol{\zeta}(t)$ to the output measurements vector $\tilde{\boldsymbol{y}}(t)$ of (3.6) is as follows*

$$\sup_{\|\boldsymbol{\zeta}(t)\|_2 \neq 0} \frac{\|\tilde{\boldsymbol{y}}(t)\|_2}{\|\boldsymbol{\zeta}(t)\|_2} = \sigma_{\max}(G(0)) = \sigma_{\max}\left(\frac{1}{k_p} C L_g^{-1} B\right), \tag{3.7}$$

where $\sigma_{\max}$ is the maximum singular value and the L_2-norm of a signal $\boldsymbol{x}$ is $\|\boldsymbol{x}\|_2^2 \triangleq \int_0^\infty \boldsymbol{x}^\mathsf{T}\boldsymbol{x} dt$. □

Proof: Taking the Laplace transform from the second row of (3.6) yields

$$s^2 \tilde{P}(s) = -k_p L_g \tilde{P}(s) - s k_v L_g \tilde{P}(s) + B Z(s), \tag{3.8}$$

where $\tilde{P}(s)$ and $Z(s)$ are the Laplace transform of $\tilde{\mathbf{P}}$ and $\boldsymbol{\zeta}(t)$, respectively. Moreover, taking the Laplace transform from the second equation of (3.6) gives

$$\tilde{Y}(s) = C\tilde{P}(s) = \underbrace{C\left(s^2 I + (k_p + s k_v) L_g\right)^{-1} B}_{G(s)} Z(s), \tag{3.9}$$

which completes the proof. ■

We define an attacker-detector game as follows. The attacker chooses f vehicles to attack such that L_2-gain from attack signal to monitoring nodes is minimized. On the other hand the detector chooses f vehicles to monitor such that L_2-gain from attack signal to monitoring nodes is maximized.

Remark 3.2. It is common in the literature that the defender (here detector) knows an upper bound of the attacked nodes [157]. Here, we assume that f is an upper bound of the attacked nodes, and hence, the detector acts based on this worst-case scenario. □

Based on Proposition 3.1, the cost function that the attacker tries to minimize and the detector tries to maximize is defined as follows

$$J(B,C)=\sigma_{\max}\left(G(0)\right)=\frac{1}{k_p}\sigma_{\max}\left(CL_g^{-1}B\right). \tag{3.10}$$

It is proved in the literature that when the graph $\mathcal{G}$ is connected (which holds for a platoon that is a line graph), then L_g is nonsingular and L_g^{-1} is nonegative elementwise [158].

Remark 3.3. The proposed approach in this chapter is basically considered as a centralized one. In particular, as in (3.10), the global knowledge of the variables of the Laplacian matrix need to be known for the game pay-off to be fully defined. The elements of L_g^{-1} are determined based on the special form of this matrix according to the exploited information flow topology. This will be explained in Lemma 3.5 and 3.6. □

The following lemma will be needed in the subsequent attacker-detector game analyses.

Lemma 3.4 ([159]). *For a non-negative matrix, A, the largest singular value is a non-decreasing function of its elements. Besides, if A is irreducible, then the largest singular value is a strictly increasing function of its entries.* □

3.3 Single Attacked–Single Detecting Vehicles

In this section, we investigate the existence of an equilibrium point of the attacker-detector game in both undirected and directed cases where there is only one attacked node. To this end, we first present explicit representations of L_g^{-1} for both of these scenarios in the two following lemmas, respectively. The proof of these results are presented in the Appendix.

Lemma 3.5. *Suppose that $\mathcal{G}_u$ is a weighted undirected path graph and let $\mathcal{P}_{i\ell}$ be the set of nodes involved in the (unique) path from the leader node v_ℓ to v_i (including v_i). Then we have*

$$[L_g^{-1}]_{ij}=\sum_{\ell\in\mathcal{P}_{i\ell}\cap\mathcal{P}_{j\ell}}\frac{1}{w_\ell}. \tag{3.11}$$

□

Lemma 3.6. *Suppose that $\mathcal{G}_d$ is a weighted directed path graph with the leader node v_ℓ. Then, the entries of the matrix L_g^{-1} are given by*

$$[L_g^{-1}]_{ij}=\begin{cases}\frac{1}{w_j} & \text{if there is a directed path from } j \text{ to } i,\\ 0 & \text{if there is no directed path from } j \text{ to } i.\end{cases} \tag{3.12}$$

□

Remark 3.7. In case $f = 1$ where the attacker attacks node j and the detector places its sensor on node i, i.e., $B = \mathbf{e}_j$ and $C = \mathbf{e}_i^\mathsf{T}$ for some $1 \leq i, j \leq n$, the game pay-off is reduced to the following simple form

$$\sigma_{\max}\left(\frac{1}{k_p} C L_g^{-1} B\right) = \sigma_{\max}\left(\frac{1}{k_p} \mathbf{e}_i^\mathsf{T} L_g^{-1} \mathbf{e}_j\right) = \frac{1}{k_p}[L_g^{-1}]_{ij}, \tag{3.13}$$

where $[L_g^{-1}]_{ij}$ is the ij^{th} element of L_g^{-1}. □

The following result presents the existence of an equilibrium point in a weighted undirected path graph.

Theorem 3.8. *Let $\mathcal{G}_u$ be a weighted undirected path graph with v_ℓ as the leader node in one end of the graph. Assume that the weight of an incoming edge from v_ℓ to node i is w_i. The game between the attacker and the detector has at least one NE and the game value is $\frac{1}{w_1}$ where w_1 is the weight of the incoming edge to the leader's neighbor node v_1.* □

Proof: The NE pertains to the scenario in which the attacker attacks the leader's neighbor node. This fact is easily derived based on Lemma 3.5. In fact, the attacker tries to minimize the game objective by attacking the nearest node to the leader so as to regardless of the detection node, the number of common nodes from the leader to the attacked and defended nodes is minimized (which will be 1 in this case). Hence, regardless of the detector's action, the game admits at least one NE with the same game value, i.e., $\frac{1}{w_1}$ where node 1 is the leader's neighbor. Besides, if the attacker chooses any other nodes, the game pay-off will be at least $\frac{1}{w_1}$. ■

Remark 3.9. In Theorem 3.8, one of the NEs happens where the detector places its sensor on the farthest node from the leader. This is a particular case on which we will focus in the rest of the chapter. □

The following result presents the existence of an equilibrium point in a platoon equipped with directed communication links modeled by a weighted directed path graph.

Theorem 3.10. *Let $\mathcal{G}_d$ be a weighted directed path graph with v_ℓ as the leader node in one end of the graph. Assume that the weight of an incoming edge from v_ℓ to node i is w_i. Then, the game between the attacker and the detector admits an NE in node $v_k = \arg\max_{i \in \mathcal{W}} w_i$.* □

Proof: Without loss of generality, let us denote the ordering of the nodes starting from the leader and ending at the end of the platoon by $v_\ell, v_1, v_2, \ldots, v_n$. Having in mind that in the case of a directed graph, L_g^{-1} is a lower-triangular matrix, we try to find the

equilibrium point of the attacker-detector game. We know that, based on Lemma 3.6, the last row of L_g^{-1} is $[L_g^{-1}]_{n,1\leq j\leq n} = \begin{bmatrix}\frac{1}{w_1} & \frac{1}{w_2} & \cdots & \frac{1}{w_n}\end{bmatrix}$. One can perceive that for the detector to maximize the game objective, he definitely chooses the last row of L_g^{-1} to assure there is no zero entry in the chosen row. On the other hand, the attacker has no other way except choosing the minimum entry of the aforementioned row. This entry corresponds to the node with the maximum incoming weight which completes the proof. ■

3.4 Attacker–Detector Game: $f > 1$ Case

Due to the availability of redundant on-board sensors on most of the vehicles from one hand, and that the attacker typically tends to attack more than one vehicle of the platoon to achieve a higher level of devastation from the other hand, it is more crucial for the detector to benefit from the sensor redundancy and be prepared for such attacks. In these attacks, the attacker targets more than one vehicle, and the detector is supposed to deploy more than one monitoring sensor. Hence, we extend our previous results and analyze the existence of an equilibrium point of the attacker-detector game in both undirected and directed cases where there are more than one attacked nodes.

The following result presents the existence of an equilibrium point in a weighted undirected path graph with multiple attacked nodes and multiple deployed sensors.

Theorem 3.11. *Let $\mathcal{G}_u$ be a weighted undirected path graph with v_ℓ as the leader node in one end of the path. Then for any $f > 1$, the attacker-detector game described by the game pay-off (3.10) admits at least one NE happening when the attacker chooses f closest nodes to the leader and the detector chooses f farthest nodes from the leader.* □

Proof: The structure of L_g^{-1} for a general undirected path graph shown in Fig. 3.1a, is as follows

$$L_g^{-1} = \begin{bmatrix} \frac{1}{w_1} & \frac{1}{w_1} & \cdots & \frac{1}{w_1} \\ \frac{1}{w_1} & \frac{1}{w_1}+\frac{1}{w_2} & \cdots & \frac{1}{w_1}+\frac{1}{w_2} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{w_1} & \frac{1}{w_1}+\frac{1}{w_2} & \cdots & \frac{1}{w_1}+\frac{1}{w_2}+\ldots+\frac{1}{w_n} \end{bmatrix} \tag{3.14}$$

Based on the specific structure of (3.14), i.e., the entries monotonically increase as we go further in rows/columns, the NE occurs when the attacker chooses the first f columns of L_g^{-1} and the detector chooses the last f rows of it. Denoting the so-called columns and rows by B^* and C^*, respectively, based on Lemma 3.4, one can easily see that (we omit

the coefficient $\frac{1}{k_p}$ for convenience)

$$\sigma_{\max}(CL_g^{-1}B^*) \leq \sigma_{\max}(C^*L_g^{-1}B^*) \leq \sigma_{\max}(C^*L_g^{-1}B), \tag{3.15}$$

where, B and C are any combination of f columns and rows of L_g^{-1}, respectively. If the attacker chooses columns corresponding to B (instead of B^*), then the elements of $C^*L_g^{-1}B$ increase (compared to $C^*L_g^{-1}B^*$) which in turn results in increasing $\sigma_{\max}(C^*L_g^{-1}B)$ (based on Lemma 3.4). Furthermore, if $n \leq 2f$, then any unilateral deviation of the detector's decision decreases $\sigma_{\max}(CL_g^{-1}B^*)$. In the case where $n > 2f$, the unilateral deviation of the detector's decision may not change the elements of $CL_g^{-1}B^*$ which results in more than one NE with the same game value. ■

The following theorem represents the existence of an equilibrium point in a weighted directed path graph with $f > 1$ attacked nodes and $f > 1$ deployed sensors.

Theorem 3.12. *Let $\mathcal{G}_d$ be a weighted directed path graph with v_ℓ as the leader node in one end of the path. Then for any $f > 1$, the attacker-detector game (3.7) admits an NE happening when the detector chooses f farthest nodes from the leader.* □

Proof: The structure of L_g^{-1} for a general directed platoon shown in Fig. 3.1b, is as follows

$$L_g^{-1} = \begin{bmatrix} \frac{1}{w_1} & 0 & \cdots & 0 \\ \frac{1}{w_1} & \frac{1}{w_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{w_1} & \frac{1}{w_2} & \cdots & \frac{1}{w_n} \end{bmatrix} \tag{3.16}$$

Based on the lower-triangular structure of (3.16), the game admits an NE when the detector chooses the last f rows of L_g^{-1}. Moreover, the attacker's decision is highly sensitive to the weight assignments. Particularly, based on the values of $w_i, 1 \leq i \leq n$, the attacker has to solve a minimization problem to achieve the least value for the game value corresponding to his strategy. Different scenarios for this decision making will be explained in Sec. 3.7. ■

Remark 3.13 (Computational cost of the attacker). In a general weighted directed platoon where there are more than one attacked nodes, the attacker's decision is highly sensitive to weight assignments since the attacker has to solve a computationally complex optimization problem. Particularly, he has to calculate the game value for every combination of selecting f columns out of n columns of the Laplacian matrix. Mathematically, the cost of this computation is evaluating the maximum singular value of the resulting $f \times f$ matrix for every $\binom{n}{f}$ selections. Depending on the values of f and n, this computation can be of high burden. □

3.5 Effects of Adding Extra Communication Links to a Platoon

In real vehicle platoons, it might be the case that additional communication links either undirected or directed are added between the vehicles. This will clearly affect the existing communication environment between the vehicles and the security level of the new platoon. Hence, in this section, we discuss the impact of adding extra links to a path graph on the security level of the resulting graph in both undirected and directed cases.

3.5.1 Undirected Case

We consider the general scenario in which an extra edge with weight w_i (modeling the added communication link) is added from node j to node i. In the undirected case, this extension can be generally formulated as follows

$$\tilde{L}_g = L_g + w_i \mathbf{e}_{ij} \mathbf{e}_{ij}^{\mathsf{T}}, \tag{3.17}$$

where, $\tilde{L}_g$ is the perturbed Laplacian matrix corresponding to the new graph, $\mathbf{e}_{ij} = \mathbf{e}_i - \mathbf{e}_j$, and $\mathbf{e}_i$ is a vector with 1 in the i^{th} position and 0 elsewhere. The following result presents the effect of adding an extra communication link between two nodes of vehicle platoon on its security level. The proof of the following theorem is presented in the Appendix.

Theorem 3.14. *Let $\mathcal{G}_u$ denote a weighted undirected path graph. Then, adding an extra edge to $\mathcal{G}_u$ will decrease the game value.* □

Theorem 3.14 indicates that adding new communication links to a platoon equipped by bidirectional communication links between the vehicles lessens the detectability (visibility) of the attack. In fact, the attack signal finds more ways to be distributed through the new links which in turn reduces its power (energy). Consequently, the attack becomes less visible and more difficult to be detected, creating a less secure platoon.

3.5.2 Directed Case

In the directed case, this extension can be generally formulated as follows

$$\tilde{L}_g = L_g + w_i \mathbf{e}_i \mathbf{e}_{ij}^{\mathsf{T}}, \tag{3.18}$$

Theorem 3.15. *Let $\mathcal{G}_d$ denote a weighted directed path graph. Then, adding an extra edge to $\mathcal{G}_d$ which makes a cycle will increase the game value and adding an extra edge to $\mathcal{G}_d$ which does not make a cycle will decrease the game value.* □

Proof: The proof will be given in the Appendix. ■

Remark 3.16. The results of this section make real sense from a practical point of view. Particularly, in a platoon equipped by unidirectional communication links (the directed case), when the extra link is added between two vehicles creating a cycle, this data flow cycle is created in which the attack signal is circulated and becomes more visible (detectable). It is worth noting that as this is a directed flow path, there is no power loss for the attacker while it is circulating. Hence, the game value, i.e., the detectability of the attacker increases. In the case where no cycle is made, there is no data flow path created for the attack to be propagated. This physically dampens the attack effect. Thus, the attacker becomes less visible in the new platoon, and naturally, the game value is decreased. The same reasoning holds for the undirected case. □

3.6 Security Level of a Platoon with Bidirectional versus Unidirectional Communication Links

In this section we briefly study the security level of a platoon equipped by either a bidirectional or unidirectional communication links. The following proposition establishes the result.

Proposition 3.17. *Let $\mathcal{G}_u$ and $\mathcal{G}_d$ denote a weighted undirected and directed vehicle platoon, respectively. The game value corresponding to the attacker-detector game of the undirected platoon is larger than the directed one, hence, is more secure.* □

Proof: Let us first consider the $f = 1$ case. Based on the general structure of L_g^{-1} for the undirected and directed cases given in (3.14) and (3.16), respectively, one can easily see that each element of (3.16) is not larger than the corresponding element of (3.14). This is basically due to the way that these matrices are formed based on Lemma 3.5 and Lemma 3.6. Now let us consider the $f > 1$ case. With the same argument, we can immediately perceive that all the elements of the $f \times f$ matrix $\left(CL_g^{-1}B\right)_{\text{directed}}$ are not larger than the corresponding elements of $\left(CL_g^{-1}B\right)_{\text{undirected}}$ for any attacker and detector decisions. This together with Lemma 3.4 complete the proof. ■

This result verifies that when a platoon is equipped by bidirectional communication links among the vehicles (the undirected case), each vehicle can send and receive more

data from its both follower and preceding vehicles. This clearly causes a more secure platoon. In the directed case, i.e., the communication links are of the unidirectional type, each vehicle is only able to receive data from its preceding vehicle, hence, the detectability of the attacker might not be maximized compared to the undirected case. Hence, the security level of the latter platoon is lower than the first one.

Fig. 3.2 shows the entire procedure to determine the optimal strategy for the detector.

3.7 Simulation and Experimental Results

3.7.1 Simulation Results

Here, the application of the aforementioned results in a vehicle platoon subject to bias injection attacks in two different cases namely, undirected and directed platoons is investigated. In the considered platoon, we place the leader at one end of the path and keep the same labeling policy for the vehicles as before.

$f = 1$ Case

In this case, we consider a weighted platoon formation in which the attacker attacks one vehicle and the detector places one sensor on a specified vehicle. This sensor placement has to be optimized based on NE of the attacker-detector game. We consider a platoon with 5 vehicles. The weights have been chosen as, $w_1 = 2, w_2 = 2.5, w_3 = 1.5, w_4 = 3$, and $w_5 = 2.75$. Fig. 3.3 shows the game values for both the undirected and directed cases where $f = 1$. For the undirected case (Fig. 3.3a), based on Theorem 3.8, the game has non-unique NEs happening in the leader's neighbor vehicle regardless of the detector's action. For the directed case (Fig. 3.3b), based on Theorem 3.10, the game has a unique NE in the vehicle with maximum incoming weight, which is w_4. From Fig. 3.3, one can easily see that in both undirected and directed cases, if the attacker chooses a vehicle other than the shown NE, the game value increases. Besides, if the detector chooses a vehicle other than the shown NE(s), the game value decreases. Hence, neither the attacker nor the detector are willing to change their strategies.

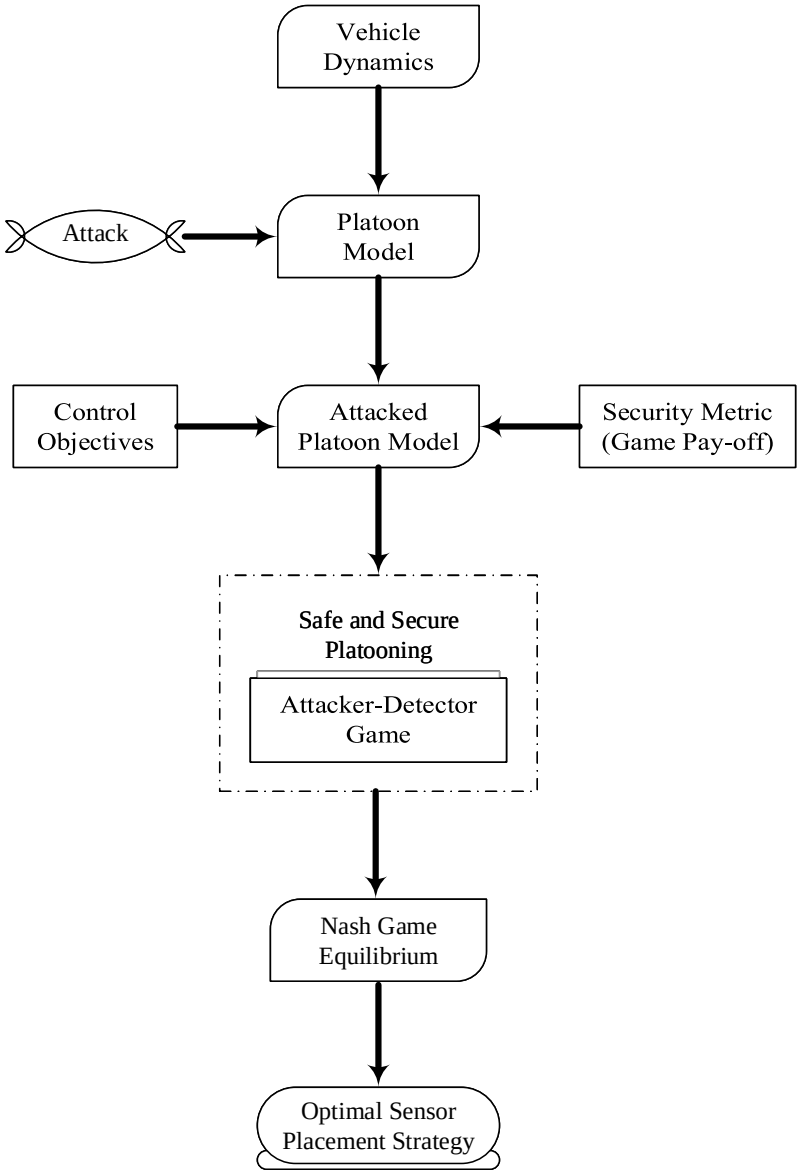
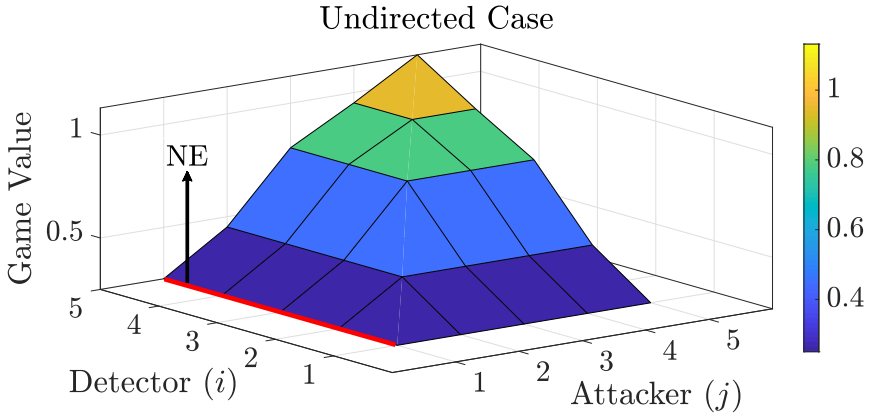


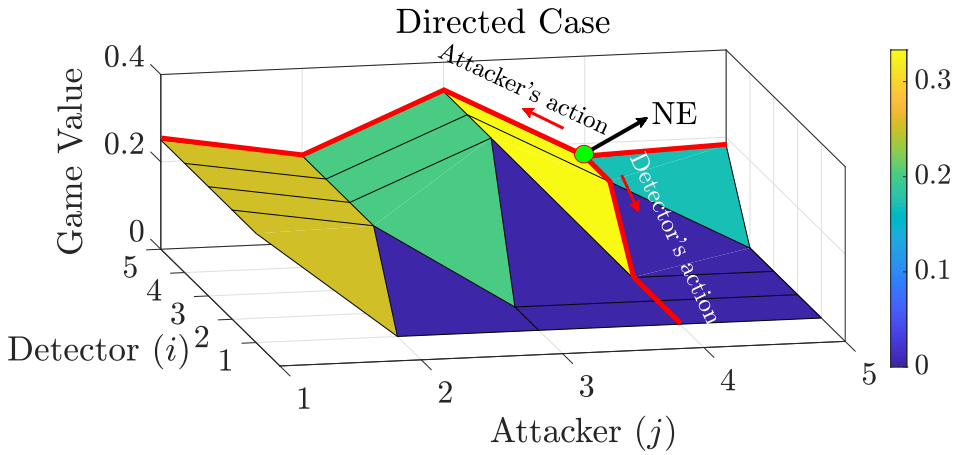
Figure 3.2: Proposed procedure to obtain the optimal strategy for the detector

$f > 1$ Case

In this case, we consider a similar platoon with 5 vehicles as in the previous case, and $f = 2$. The weights w_1 through w_5 are the same as before. In this case, the attacker



(a) Game values for a weighted undirected platoon with $f = 1$



(b) Game values for a weighted directed platoon with $f = 1$

Figure 3.3: Game values and NE for weighted undirected and directed platoons for $f = 1$

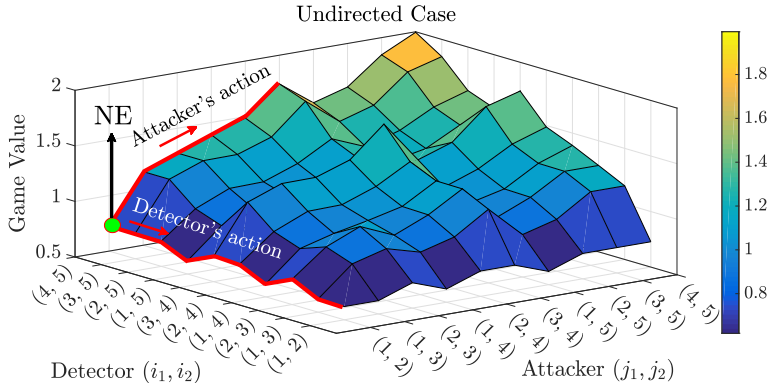
attacks a pair of vehicles (j_1, j_2) and the detector places its sensors on a pair of vehicles (i_1, i_2). Fig. 3.4 shows the game values for the undirected case where $f = 2$. According to Theorem 3.11, the game admits at least one NE where the attacker attacks 2 closest vehicles to the leader, and the detector chooses the 2 farthest vehicles from the leader. In this case, since $n > 2f$, the game has non-unique NEs. These NEs occur when the attacker attacks 2 closest vehicles to the leader while the detector can choose any pair of vehicles such that they do not include the leader's neighbor vehicle. Fig. 3.4 shows one specific NE where the attacker attacks the pair $(1, 2)$ (two closest vehicles to the leader) and the detector chooses the two farthest vehicles from the leader, which is the pair $(4, 5)$. It is

easily seen that neither the attacker nor the detector are willing to change their actions.

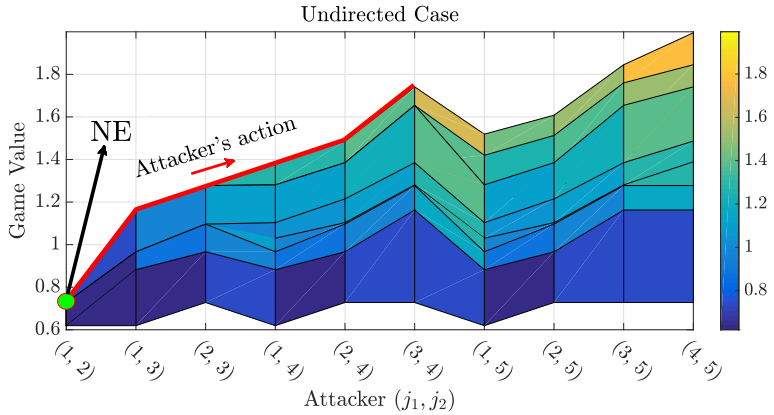
In the directed scenario, based on Theorem 3.12, there exists an NE which happens when the detector places two sensors in the farthest vehicles from the leader. Fig. 3.5 shows the game values for this scenario in which the game admits an NE where both the attacker and the detector choose the 2 farthest vehicles from the leader.

For the directed case, we present the following example showing that the attacker's decision is highly sensitive to weight assignments.

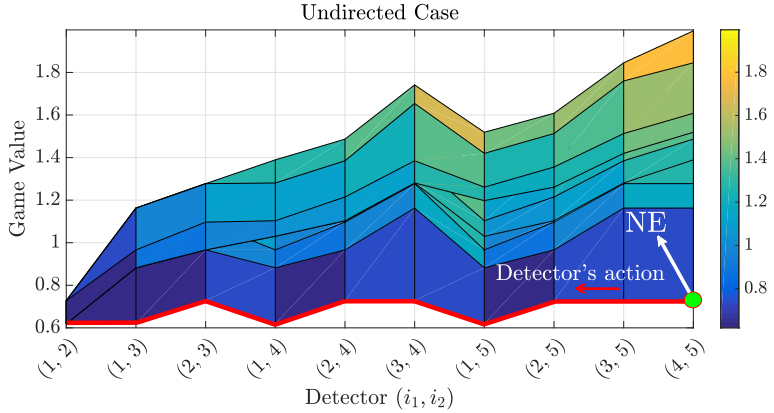
Example 3.18. Consider the weighted directed platoon shown in Fig. 3.1b with $n = 5$ vehicles, $f = 2$, and the following weights, $w_1 = 2000, w_2 = 0.1, w_3 = 0.05, w_4 = 0.1$, and $w_5 = 0.01$. Based on Theorem 3.12, there exists an NE where the detector chooses the 2 farthest vehicles from the leader. Fig. 3.6 shows the game value for this example. In this specific platoon, one can easily see that the game admits an NE which happens when the attacker attacks the 2 closest vehicles to the leader. Based on the lower-triangular structure of the Laplacian matrix, although there exists a zero element in the fifth column of L_g^{-1}, the attacker is willing to choose the first two columns (not choosing the fifth one at all) to achieve a lower game value. This example clearly verifies the high dependency of the attacker's decision on the weight assignments. □



(a) General view

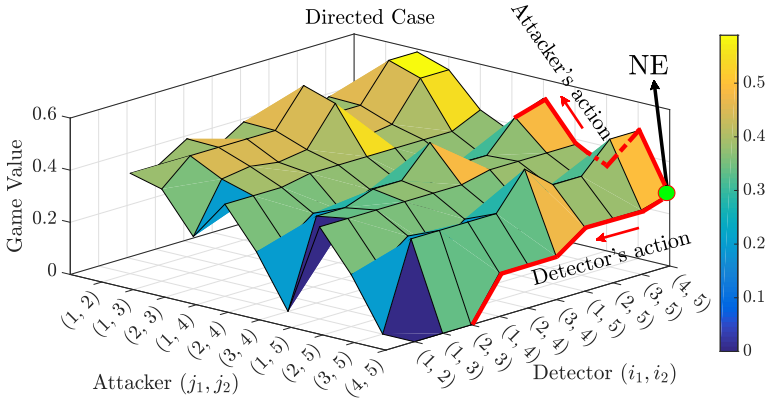


(b) Attacker view

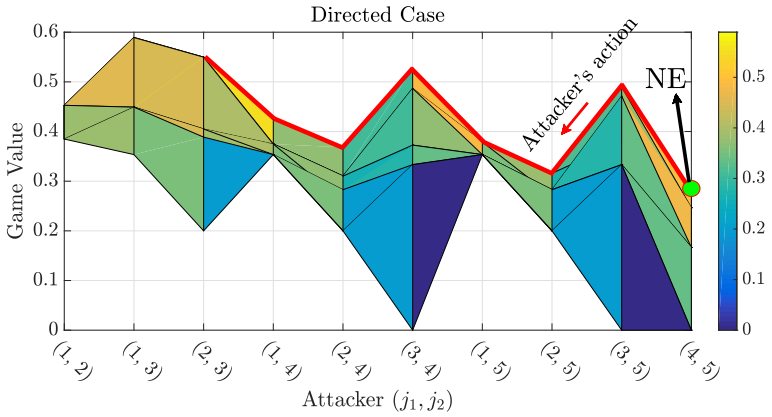


(c) Detector view

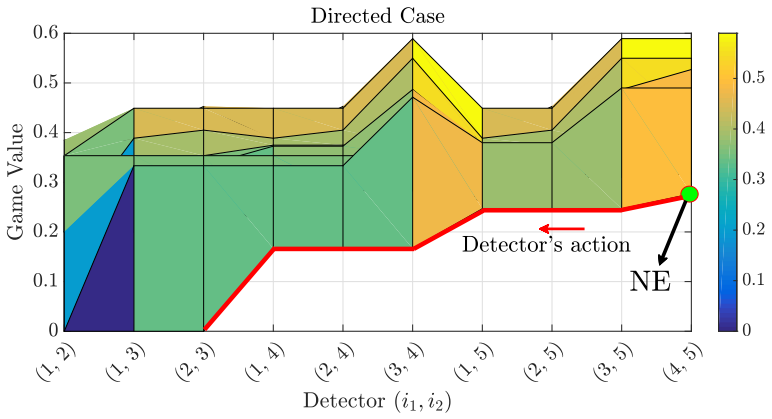
Figure 3.4: Game values and NE for a weighted undirected platoon with 5 vehicles and $f = 2$



(a) General view



(b) Attacker view



(c) Detector view

Figure 3.5: Game values and NE for a weighted directed platoon with 5 vehicles and $f = 2$

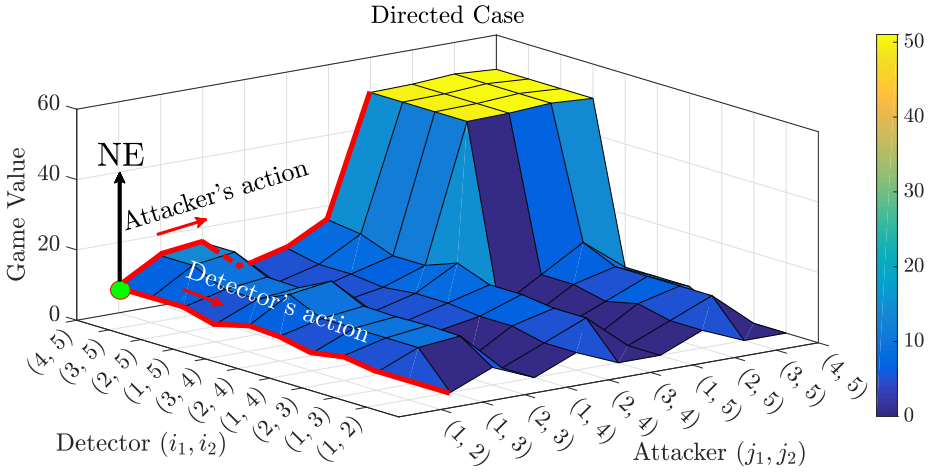


Figure 3.6: Game values for the specific weighted directed platoon in Example 3.18

3.7.2 Experimental Results

We have conducted experimental tests on a vehicle platoon setup operated by Robotic Operating System (ROS).

Experimental Setup Configuration

The setup is consisted of 3 vehicles driving on a treadmill (see Fig. 3.7). The vehicles' positions are captured by a central infrared camera detecting the specific Apriltags mounted on the vehicles. Here we consider a virtual leader specifying a desired speed profile, generated by the host PC, with 3 followers that have to follow this common profile. Each of the vehicles is equipped by a cascaded PID controller which commands the vehicle to follow the leader's speed profile and keep the desired safe distance with its preceding vehicle. The control signals are commanded based on the received data from the central ROS run on the host PC. In this setup, the data transfer between the vehicles is of directed predecessor-follower type, i.e., each vehicle can receive data from its predecessor. The data is exchanged between the host PC running the ROS and the vehicles through an IEEE 802.15.4-based 2.4GHz ZigBee wireless network protocol (see Fig. 3.8). The position, linear and angular velocity, steering and the throttle of the vehicles are measured in real-time via the ROS. Two different attack scenarios, namely an acceleration-brake attack and a brake-acceleration attack will be generated, and the results confirming our theoretical analyses

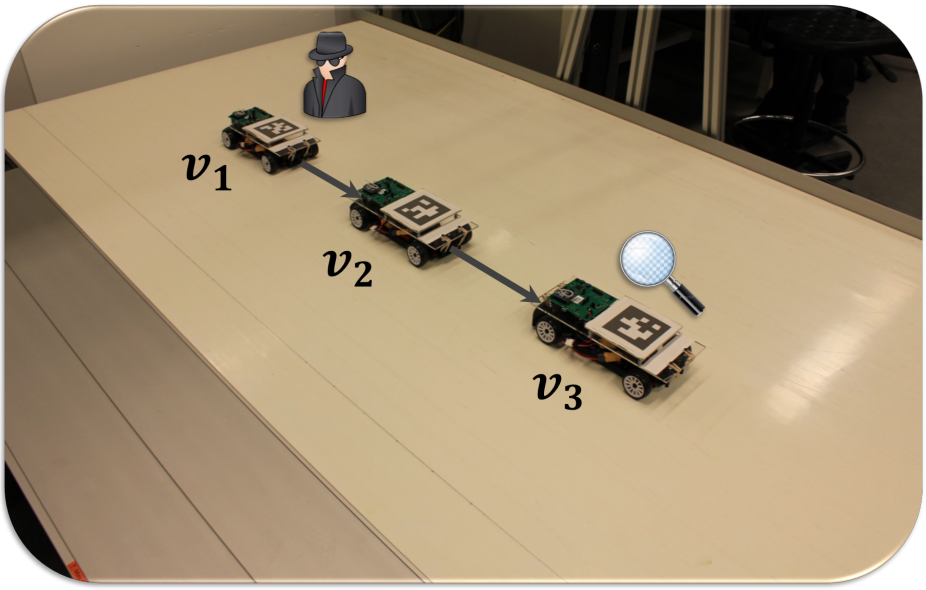


Figure 3.7: Vehicle platoon experimental setup

will be demonstrated. It is noteworthy that our experimental results are in line with the string stability notion in vehicle platoons as well [160].

Attack Scenario I (Acceleration–Brake Attack)

In this experiment, we attack the first follower (vehicle v_1) by an acceleration followed by a brake (see Fig. 3.9). Hence, at the beginning, this vehicle accelerates forward and gets far from its desired position and then gets back to its original position. Subsequently, the other two followers have to accelerate first and then brake to keep the desired inter-vehicular distance among the platoon. Fig. 3.10 demonstrates the position error and the 2-norm of error signals for the followers. From Fig. 3.10a, it is obviously seen that the attack effect has been propagated through the upstream of the platoon with a time delay. Fig. 3.10b shows that the norm of the error signal of the last follower will eventually get the highest value in a finite time. As the norm of the attack signal is a fixed value, based on Proposition 3.1, the game value (detectability of the attacker) will get the highest value if the last vehicle in the platoon is monitored. Hence, the detector has to place his monitoring sensor on the last follower to increase the security level of the system. This clearly confirms our result for the detector strategy presented in Theorem 3.10.

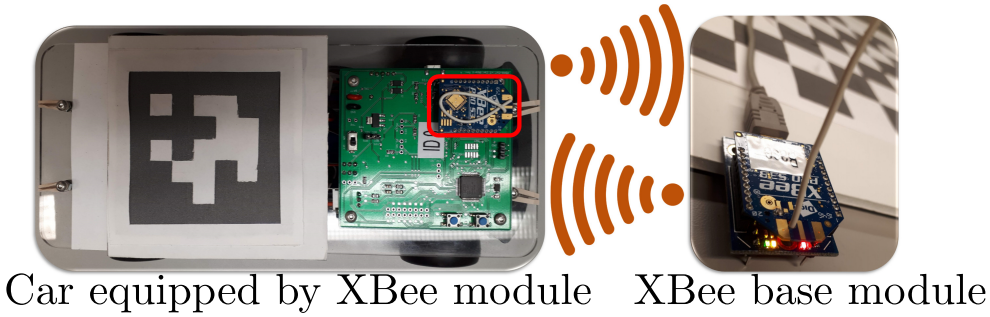


Figure 3.8: XBee network connection

Attack Scenario II (Brake–Acceleration Attack)

In this scenario, we attack the second follower (vehicle v_2) by forcing it to have a brake followed by an acceleration (see Fig. 3.11). In a real platoon, this kind of attack could be of high significance as it can result in severe braking of the other followers resulting in a huge degradation of the driving comfort and safety. Fig. 3.12 shows the position error and the 2-norm of the error signals for the vehicles. As it can be seen from Fig. 3.12a, due to the unidirectional data transfer in the platoon, the attack occurred on v_2 does not affect v_1. Again the effect of the attack on v_2 propagates to v_3 with a short time delay. From Fig. 3.12b, similar to the first scenario, the detectability of the attacker is maximized if the detector places its sensor on the last follower which again verifies our previous theoretical results.

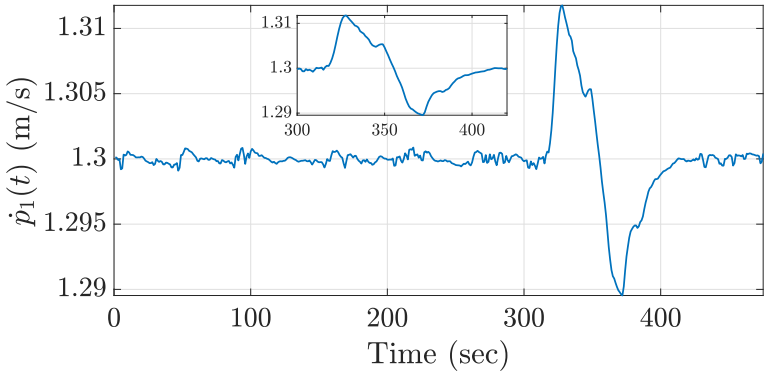
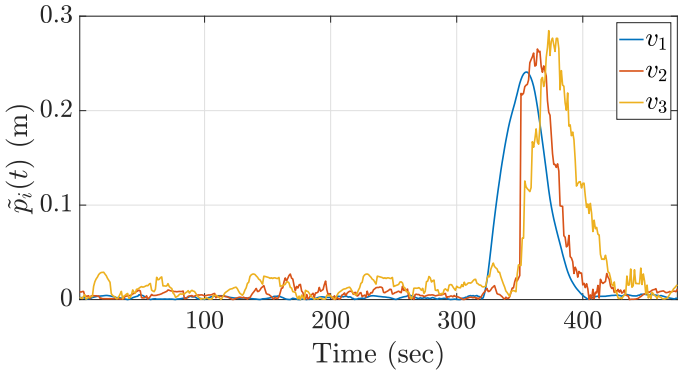
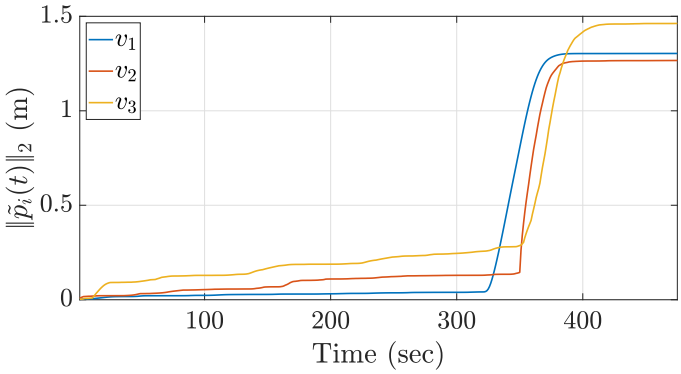


Figure 3.9: Velocity of the attacked car in scenario I



(a) Position error of the vehicles in scenario I



(b) Norm of the measurement signals in scenario I

Figure 3.10: Position error and norm of the measurement signals of the follower vehicles in scenario I

Remark 3.19. We can see from Fig. 3.9 and 3.11 that the attacks occurred in about 100 and 80 seconds in scenario I and II, respectively. Having been approximated the acceleration followed by a brake in scenario I (and the brake followed by an acceleration in scenario II) with a sinusoidal signal, they reflect approximately 0.01 – 0.02 Hz attacks. Hence, they can be reasonably captured as the low-frequency attacks. □

Remark 3.20. It is worth mentioning that according to our analytical results (Theorem 3.10), in a weighted directed platoon, the optimal strategy for the detector is to choose the farthest vehicle from the leader to place the monitoring sensor. Since in our experimental results the quality of the communication links among the vehicles are the same (the weights are all equal), the game admits more than one NE with the same game pay-off regardless of the attacker's action. This was the situation in the scenario I and II where different vehicles of the platoon were attacked. □

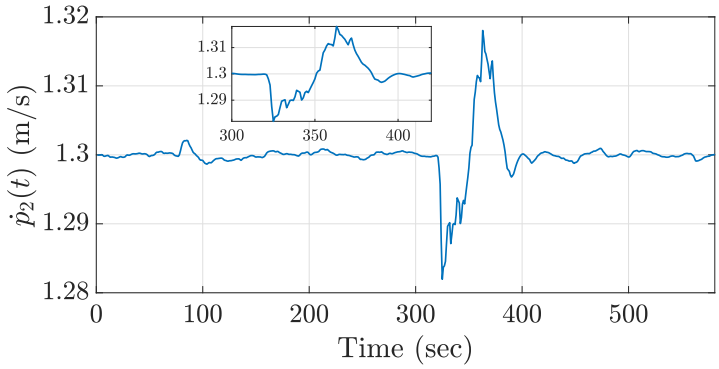
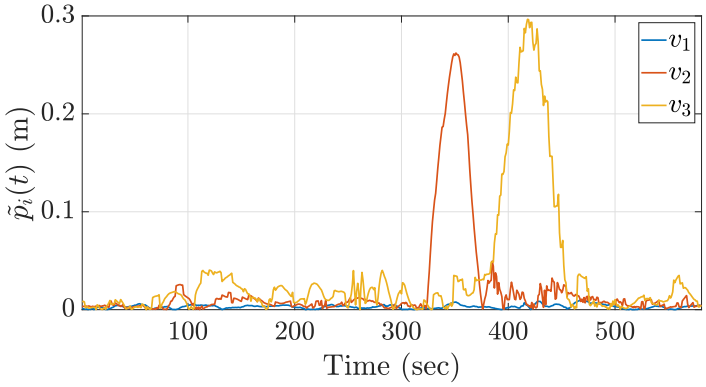
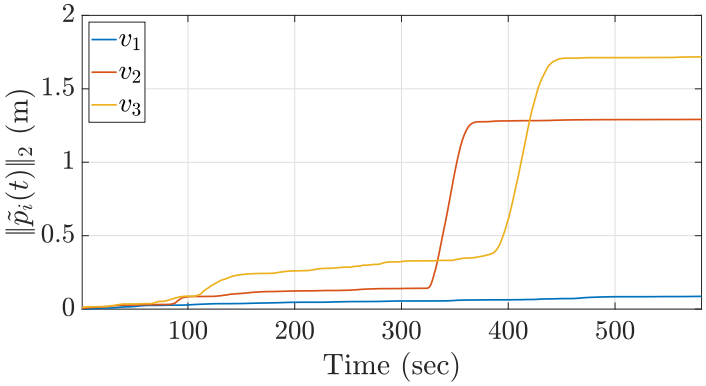


Figure 3.11: Velocity of the attacked car in scenario II



(a) Position error of the vehicles in scenario II



(b) Norm of the measurement signals in scenario II

Figure 3.12: Position error and norm of the measurement signals of the follower vehicles in scenario II

Remark 3.21. There are some limitations regarding the experimental setup used in this work. To point out some of the most important limitations, we notice that the wireless communication among the scaled cars is fast enough and transfers data with relatively low latency, while in the real scenario, there could be significant time delay in data transfer. Besides, they do not bear large inertia, and the friction force among the wheels and the treadmill is negligible. Therefore, it is reasonable to capture their dynamics via a linear model. These limitations have to be taken into account while implementing the method on real vehicles. □

3.8 Summary

In this chapter, we have focused on security and robustness analysis of vehicle platoons based on a graph-theoretic approach. The vehicles have been assumed to be able to communicate data, such as inter-vehicular distance and speed among each other via wireless communication environments. Both the unidirectional and bidirectional data transfer have been studied. Moreover, the quality of communication links between the vehicles has been considered using edge weights of the underlying path graph topology. The platoon is assumed to be under cyber attacks, and a detector is supposed to choose a strategy to place his monitoring sensors on specific vehicles aiming at increasing the detectability of the attacker. An attacker-detector game has been defined based on which the existence of any possible NE points have been studied. Based on our results, the detector can decide about his sensor placement strategy to increase the security level of the system. Also, robustness analysis of a platoon against adding extra communication links between the vehicles has performed. Furthermore, our study verifies the fact that using a bidirectional communication environment forms a more secure platoon compared to the unidirectional counterpart. Our simulation and experimental results verified the effectiveness of our theoretical analyses. An open avenue for the current research is to extend the underlying graph topology such that it can handle dynamic platoon formations resulted from different vehicle maneuvers such as cut-in/cut-out actions, hence, studying the impacts of those movements on the security of vehicle platooning. Besides, the extension of this work to the dynamic game (with changing network topology) along with generalizing our method for possibly different vehicle dynamics in the platoon referred to as the "heterogeneous" case are left as our future studies.

Chapter 4

Security-Aware Optimal Actuator Placement in Vehicle Platooning

In this chapter, we deal with the security challenges of vehicle platoons equipped by distributed consensus controllers under the risk of cyber attacks. In particular, we mainly focus on the h–nearest neighbor platoons benefiting from either unidirectional or bidirectional data communications. We let the attacker be able to inject acceleration attack signals to the longitudinal dynamics of one (or more) of the vehicles present in the platoon and try to infer the best defense strategy to mitigate the attack effects. We propose a general approach to find an optimal actuator placement strategy according to the Stackelberg game between the attacker and the defender. The game payoff is the energy needed by the attacker to steer the consensus follower-leader dynamics of the system towards his desired direction. The attacker tries to minimize this energy while the defender attempts to maximize it. Thus, based on the defined game and its optimal equilibrium point, the defender(s) selects optimal actuator placement action to face the attacker(s). Both cases of single attacker–single defender and multiple attackers–multiple defenders cases are investigated. Furthermore, we study the effects of different information flow topologies, namely the unidirectional and bidirectional data transfer structures. Besides, the impacts of increasing the connectivity among the nodes on the security level of the platoon are presented. Simulation results for h–nearest neighbor platoon formations along with experimental results using the scaled cars governed by Robotic Operating System (ROS) verify the effectiveness of the method. Explicitly, the contributions of current chapter are as follows,

- Considering the longitudinal dynamics of a vehicle, we formulate the attacker-defender

game as a Stackelberg game problem (4.15) with the attacker(s) and the defender(s) as the game players. Both of the single attacker–single defender and multi attackers–multi defenders scenarios will be investigated. Furthermore, we study the impact of different information flow topologies, namely the unidirectional and bidirectional data transfer structures on the security level of the system.

- Two energy-related game pay-offs are introduced, namely the trace and the largest eigenvalue of the controllability Gramian matrix. The former has been introduced in complex dynamics networks literature [161, 162] and the latter is introduced in this work. Their interpretation together with their usefulness for our problem is described.

- A general algorithm to solve the Stackelberg game problem is given, whereby the optimal solution of the game determines the optimal actuator placement strategy for the defender. The proposed approach can be applied to platoon formations with arbitrary information flow topologies.

- Based on the values of the game-payoffs, the effect of increasing the connectivity among the vehicles of the platoon on its security level is demonstrated. This has a significant role from the defender's perspective who tries to mitigate the attack effects as much as possible.

It is notable that in the previous chapter we proposed Nash game solutions for sensor placement problem in vehicle platooning to detect attack effects and increase the security level of the system. The current chapter is basically different in terms of the inherent problem, i.e., optimal actuator placement to defend and mitigate the attack effects and also the game formulation, which is the Stackleberg game. In addition, in this work the platoon model has been generalized to a higher order model to capture more realistic platoon dynamics.

The results of this chapter have been published in [163].

4.1 Organization of the Chapter

The remainder of this chapter is organized as follows. Sec. 4.2 defines the problem statement including the longitudinal vehicle dynamics, objective of the platoon control, considered spacing policy, and the employed distributed controller. The attack model is also given in this section. Sec. 4.3 details the defined attacker-defender game along with the

algorithm to solve for the equilibrium point of the game. Simulation results on h–nearest neighbor platoons are demonstrated in Sec. 4.4 showing the effectiveness of the method. Sec. 4.5 is devoted to the experimental results. The proposed method is implemented on a platform which creates a real platoon composed of the scaled cars governed by the ROS. Finally, conclusions and open avenues to continue this work are presented in Sec. 4.6.

4.2 Problem Statement

4.2.1 System Model

We consider a platoon consisting of n follower vehicles each of which modeled with the following nonlinear dynamics model

$$\begin{cases} \dot{p}_i(t) = v_i(t), \\ \dot{v}_i(t) = \dfrac{1}{M}\left(\eta_T \dfrac{T_i(t)}{R_w} - C_A v_i^2 - Mgf_r\right), \ i = 1, 2, \ldots, n \\ \tau \dot{T}_i(t) + T_i(t) = T_{i,\text{des}}(t), \end{cases} \tag{4.1}$$

where $p_i(t)$ and $v_i(t)$ are the position and velocity of vehicle i, respectively. M denotes the vehicle mass, C_A is the coefficient of aerodynamic drag, f_r is the coefficient of rolling resistance, g is the gravity constant, $T_i(t)$ is the actual driving/braking torque applied to the drivetrain, $T_{i,\text{des}(t)}$ denotes the desired driving/braking torque, R_w is the tire radius, τ is the inertial lag of vehicle powertrain, and η_T is the mechanical efficiency of the driveline. It is assumed that the leader tracks a constant speed reference trajectory, i.e., $a_0(t) = 0$, $p_0 = v_0 t$ [164], where $p_0(t)$, $v_0(t)$, and $a_0(t)$ denote the position, velocity, and acceleration of the leader, respectively. The position output with relative degree three along with the following feedback linearization technique, which is widely used in literature [62, 164–166], are utilized to convert (4.1) to a linear one

$$T_{i,\text{des}}(t) = \frac{1}{\eta_T}\left(C_A v_i(2\tau \dot{v}_i + v_i) + Mgf_r + Mu_i\right) R_w, \tag{4.2}$$

where u_i is the control input applied to the system after feedback linearization.

Now let's define

$$a_i = \frac{1}{M}\left(\eta_T \frac{T_i(t)}{R_w} - C_A v_i^2 - Mgf_r\right). \tag{4.3}$$

Then we have

$$\begin{aligned}\tau\dot{a}_i + a_i &= \tau\frac{1}{M}\left(\eta_T\frac{\dot{T}_i(t)}{R_w} - 2C_A v_i\dot{v}_i\right) + \frac{1}{M}\left(\eta_T\frac{T_i(t)}{R_w} - C_A v_i^2 - Mgf_r\right)\\ &= \frac{\eta_T}{MR_w}\left(\tau\dot{T}_i(t) + T_i(t)\right) - \frac{1}{M}(2C_A\tau v_i\dot{v}_i - C_A v_i^2 - Mgf_r)\\ &= \frac{\eta_T}{MR_w}T_{i,\text{des}} - \frac{1}{M}(2C_A\tau v_i\dot{v}_i - C_A v_i^2 - Mgf_r).\end{aligned} \tag{4.4}$$

Substituting (4.2) into (4.4) leads to

$$\tau\dot{a}_i + a_i = \frac{1}{M}(C_A v_i(2\tau\dot{v}_i + v_i) + Mgf_r + Mu_i) - \frac{1}{M}(2C_A\tau v_i\dot{v}_i - C_A v_i^2 - Mgf_r) = u_i. \tag{4.5}$$

For the purpose of platoon control, the following 3$^{\text{rd}}$-order state space model is yielded for the i^{th} vehicle

$$\mathcal{P}: \begin{cases}\dot{p}_i(t) = v_i(t),\\ \dot{v}_i(t) = a_i(t),\\ \dot{a}_i(t) = -\dfrac{1}{\tau}a_i(t) + \dfrac{1}{\tau}u_i(t),\end{cases} \tag{4.6}$$

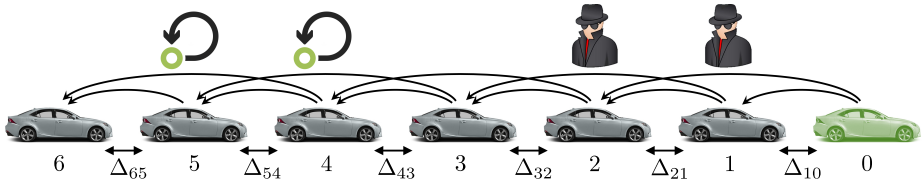
where $a_i(t) = \dot{v}_i(t)$ is the actual acceleration of the i^{th} vehicle.

In this work, we consider a vehicle platoon wherein vehicles are connected through an h–nearest neighbor information flow topology (see Fig. 4.1). This topology has been widely utilized in automotive research community. In a directed h–nearest neighbor data transfer structure, each vehicle has a look-ahead data transfer communicating with its h predecessors (Fig. 4.1a). In an undirected h–nearest neighbor data transfer structure, each vehicle looks ahead and back and exchanges vehicular data with its h followers and predecessors (Fig. 4.1b).

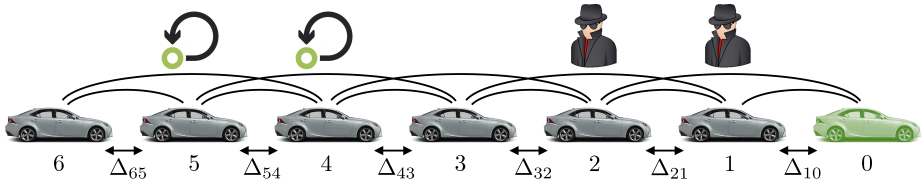
4.2.2 Control Objectives

The platoon control objective is for the followers to track the reference speed profile generated by the leader while maintaining a constant distance between any two consecutive vehicles, i.e.

$$\begin{cases}\lim\limits_{t\to\infty}\|v_i(t) - v_0(t)\| = 0,\\ \lim\limits_{t\to\infty}\|p_{i-1}(t) - p_i(t) - \Delta_{i-1,i}\| = 0,\end{cases} \quad i = 1, 2, \ldots, n, \tag{4.7}$$



(a) Unidirectional information flow



(b) Bidirectional information flow

Figure 4.1: 2-nearest neighbor platoons with different information flow topologies with sample attackers and actuators

where $\Delta_{i-1,i}$ is the desired constant space between consecutive vehicles $i-1$ and i.

The desired rigid vehicle formation will be formed by the specific constant distance Δ_{ij} between vehicles i and j, which should satisfy $\Delta_{ij} = \Delta_{ik} + \Delta_{kj}$ for vehicles i, j, and k. Considering the fact that each vehicle i has access to its own position, the positions, velocities and accelerations of its neighbors, and the desired inter-vehicular distances Δ_{ij}, we benefit from the following consensus control law, which is a distributed CACC, and is a more advanced version of the control law introduced in [155]

$$u_i(t) = \sum_{j \in \mathcal{N}_i} k_p \left(p_j(t) - p_i(t) + \Delta_{ij} \right) + k_v \left(v_j(t) - v_i(t) \right) + k_a \left(a_j(t) - a_i(t) \right), \tag{4.8}$$

By collecting the states as $x_i(t) = [p_i(t), v_i(t), a_i(t)]^\mathsf{T}$, and defining the tracking error vector for the i^{th} vehicle, $\tilde{x}_i(t) = [\tilde{p}_i(t), \tilde{v}_i(t), \tilde{a}_i(t)]^\mathsf{T} = x_i(t) - x_0(t) - \tilde{b}_i$ with $\tilde{b}_i = [\Delta_{i0}, 0, 0]^\mathsf{T}$, we get the following error dynamics model

$$\dot{\tilde{\boldsymbol{x}}}(t) = \underbrace{\begin{bmatrix} \mathbf{0}_n & I_n & \mathbf{0}_n \\ \mathbf{0}_n & \mathbf{0}_n & I_n \\ -\frac{k_p}{\tau} L_g & -\frac{k_v}{\tau} L_g & -\frac{k_a}{\tau} L_g - \frac{1}{\tau} I_n \end{bmatrix}}_{A_c} \tilde{\boldsymbol{x}}(t) \tag{4.9}$$

where $\tilde{\boldsymbol{x}} = [\tilde{\mathbf{P}} \quad \dot{\tilde{\mathbf{P}}} \quad \ddot{\tilde{\mathbf{P}}}]^\mathsf{T} = [\tilde{p}_1, \tilde{p}_2, \ldots, \tilde{p}_n, \dot{\tilde{p}}_1, \dot{\tilde{p}}_2, \ldots, \dot{\tilde{p}}_n, \ddot{\tilde{p}}_1, \ddot{\tilde{p}}_2, \ldots, \ddot{\tilde{p}}_n]^\mathsf{T}$. L_g is the grounded Laplacian matrix associated with the underlying graph topology of the platoon which is

the reduced Laplacian matrix by removing the row and the column corresponding to the leader node. Matrix A_c in (4.9) can also be rewritten as follows

$$A_c = A \otimes I_n - \left(E\mathcal{K}^{\mathsf{T}}\right) \otimes L_g \tag{4.10}$$

where $A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -\frac{1}{\tau} \end{bmatrix}$, $E = \begin{bmatrix} 0 \\ 0 \\ \frac{1}{\tau} \end{bmatrix}$, $\mathcal{K} = [k_p \;\; k_v \;\; k_a]^{\mathsf{T}}$, and $\otimes$ is the Kronecker product.

4.2.3 Attack Model

In this chapter we consider a platoon under the risk of a cyber attack imposed by an intelligent intruder on one or several vehicles. It is assumed that the attacker injects an acceleration attack to the longitudinal vehicle dynamics (4.6). The following attacked vehicle dynamics model is used in the rest of the chapter

$$\mathcal{P}_a : \begin{cases} \dot{p}_i(t) = v_i(t), \\ \dot{v}_i(t) = a_i(t) + \zeta_i(t), \\ \dot{a}_i(t) = -\dfrac{1}{\tau} a_i(t) + \dfrac{1}{\tau} u_i(t), \end{cases} \tag{4.11}$$

where $\zeta_i(t)$ is the injected attack acceleration signal. We assume that the leader is securely protected and can not be attacked by the intruder, hence, is able to follow the constant speed reference trajectory.

In order to mitigate the attack effects, one (or more defenders) are assumed to place self-feedback loops on one (or more) of the vehicle(s). This state-feedback controller uses the velocity of the defended vehicle(s). This technique has been introduced in literature aiming at a consensus resilient networked control system [167, 168]. Each defender will place a self-feedback loop with gain k (see Fig. 4.1). This defense mechanism can be formulated and integrated with the distributed CACC controller (4.8) as follows which we refer to as $\text{CACC}_{\text{defender}}$

$$u_i(t) = \sum_{j \in \mathcal{N}_i} k_p \left(\tilde{p}_j(t) - \tilde{p}_i(t)\right) + k_v \left(\tilde{v}_j(t) - \tilde{v}_i(t)\right) + k_a \left(\tilde{a}_j(t) - \tilde{a}_i(t)\right) - k\tilde{v}_i(t), \tag{4.12}$$

where k is the positive gain of the self-loop feedback from the velocity of the vehicle. This structure generally describes the formation control of autonomous agents [169].

Aggregating the defended CACC controller (4.12) with (4.11) in a matrix form yields the following closed-loop error dynamics

$$\dot{\tilde{\boldsymbol{x}}}(t) = \underbrace{\begin{bmatrix} \mathbf{0}_n & I_n & \mathbf{0}_n \\ \mathbf{0}_n & \mathbf{0}_n & I_n \\ -\frac{k_p}{\tau} L_g & -\frac{k_v}{\tau} L_g - \frac{K}{\tau} & -\frac{k_a}{\tau} L_g - \frac{1}{\tau} I_n \end{bmatrix}}_{A_a} \tilde{\boldsymbol{x}}(t) + \underbrace{\begin{bmatrix} \mathbf{0}_n \\ B \\ \mathbf{0}_n \end{bmatrix}}_{B_a} \boldsymbol{\zeta}(t), \tag{4.13}$$

where $\boldsymbol{\zeta}(t)$ is the attack vector, $K = kD_y$, k is the gain of self-loop feedbacks from the speed of the vehicles, and D_y is a binary diagonal matrix specifying the node(s) on which an actuator is placed through a self-feedback loop, i.e., the i^{th} diagonal element of D_y is 1 if the i^{th} vehicle has a self-loop and is 0, otherwise. Matrix $B = [\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_f]$ specifies f node(s) selected by the attacker. Note that matrix A_a in (4.13) can also be rewritten as follows

$$A_a = A \otimes I_n - \left(E\mathcal{K}^{\mathsf{T}}\right) \otimes L_g - \left(E\mathbf{e}_2^{\mathsf{T}}\right) \otimes K \tag{4.14}$$

4.3 Attack Effects Mitigation via Optimal Actuator Placement

In this section, we first formulate the problem as an attacker-defender game and describe its components in detail. The game-payoff, the players, and the decision variables of each of the players is explained. Then, some basics of the method, along with the details of the proposed approach, will be presented.

4.3.1 Attacker-Defender Game

To study the confrontation of the attacker and the defender in a platoon, we formulate the problem as a game with the attacker and the defender as its players. The conflicting decision between the players arises by optimizing a game pay-off in opposite directions. We will focus on two well-known metrics describing the energy needed by the attacker to deviate the dynamics of the followers towards a direction in the state space. The game pay-offs will be explained in the next subsection. The decision variable of the attacker is the B matrix, and the defender's decision variable is the diagonal matrix K in (4.13). In fact, the attacker chooses the B matrix to determine the vehicle(s) to attack attempting to minimize the energy needed to steer the consensus dynamics in the state space. On the

other hand, the defender makes a decision about the K matrix to equip certain vehicle(s) with the self-loops to defend making the energy required by the attacker as large as possible. The attacker-defender game is cast into a Stackelberg game formulation with the defender acting as the game leader. This formulation leads to study the optimal actuator placement for the defender based on the optimal equilibrium strategy of the defined game. More rigorously, the equilibrium point of the Stackelberg game determines the optimal decision for the defender by which he determines which vehicle(s) to place the actuator(s) on. Through this placement, the attack energy needed by the attacker to steer the system into his desired direction will be maximized; hence, the attack effects will be mitigated. Throughout the chapter, it is assumed that the defender has as many actuators available as the number of attackers. The results can be simply generalized to more general cases. We formally introduce the following game

Attacker–Defender Game

The attacker injects the attack signal $\zeta_i(t)$ to f vehicles to minimize his required energy (defined by one of its physical interpretations) to steer the consensus dynamics of the system towards his desired direction in the state space, while the defender places his actuators on f vehicles to maximize the energy needed by the attacker. Hence, this zero-sum game is represented by either of the two following game pay-offs

$$J(B,K) = \lambda_{\max}\left(W_c(B,K)\right), \tag{4.15a}$$
$$J(B,K) = \mathbf{tr}\left(W_c(B,K)\right), \tag{4.15b}$$

where the attacker's decision is matrix B to maximize $J(B,K)$ and the defender's decision determines matrix K to minimize $J(B,K)$ since the game pay-offs are inversely related to the average amount of energy.

Remark 4.1. It is common in the literature that the defender knows an upper bound of the attacked nodes [157]. Here, we assume that f is an upper bound of the attacked nodes, and hence, the defender acts based on this worst-case scenario. □

The game pay-offs used in (4.15) are defined in the following subsection.

4.3.2 Game Pay-off Definition and Interpretation for the Actuator Placement Problem

There have been various metrics introduced in literature with different physical interpretations related to system controllability for complex dynamical networks [161, 162]. Having performed an attack on a real system, the attacker usually needs to take an energy limit action. Hence, controllability metrics dealing with the amount of input energy required to impose the attack are of our interest. Thus, we focus on two of these energy-related metrics which are widely used to quantify the controllability level of a network, namely the largest eigenvalue of the *controllabilty Gramian* matrix, and its trace. It is notable that, due to the special control input of our system, which is the attack signal, we utilize the largest eigenvalue of the controllability Gramian matrix instead of the smallest one, which is proposed in classical control literature [161, 162]. This will be explained in more detail subsequently.

Largest Eigenvalue of the Controllability Gramian Matrix ($\lambda_{\max}(W_c)$)

The maximum eigenvalue of W_c is a worst-case metric inversely related to the amount of energy required to move the system in a direction in the state space that is the easiest to control. The eigenvector corresponding to the maximum eigenvalue of W_c is the direction in the state space that is the easiest to move the system. We intend to minimize $\lambda_{\max}(W_c)$ to maximize the control effort needed by the attacker so that he needs much energy to move the system in the easiest direction he has in hand.

Remark 4.2. Since the control input of our system is the attack signal (undesired input), we benefit from the largest eigenvalue of W_c. As this eigenvalue and its associated eigenvector pertain to the easiest direction in the state space to which the attacker can deviate the system, we target minimizing this quantity in order to make the effort needed by the attacker as large as possible. In other words, our goal is to make the system as less controllable as possible for the easiest direction the attacker has in hand. Note that in typical systems, the control input is the desired signal which imposes using the smallest eigenvalue of W_c to measure the controllability of the system from the perspective of that particular input [161, 162]. □

Trace of the Controllability Gramian Matrix ($\mathbf{tr}(W_c)$)

Trace of the controllability Gramian matrix is inversely related to the average amount of the energy required by the attacker to move the system around in the state space. We

intend to minimize $\mathbf{tr}(W_c)$ to maximize the control effort needed by the attacker.

Remark 4.3. Minimization of the trace of the controllability Gramian matrix might lead to an uncontrollable system, such that $W_c \not\succ 0$, [161]. In this case, the defender's action is to maximize the energy required by the attacker to steer the system in the controllable subspace, i.e., range(W_c). □

4.3.3 Stackelberg Game Formulation

Before delving into the game formulation, we need to highlight the reason of working with Stackelberg game. It is due to the nature of our problem (which is a design problem), and we are interested in designing the optimal actuator placement in an offline fashion. In this setup, the Stackelberg game suits better. In other words, the design problem is generally considered as a passive problem which can be reasonably captured by a Stackelberg game (the reader is referred to [147] specially Table II therein for more details). Besides, in most of the security problems, owing to the existence of the leader and follower (where one of the players has the ability to enforce his strategy on the other), it is turned out that the Stackelberg game is more suitable to formulate the problem.

Considering either of the two aforementioned metrics as our game pay-off, denoted by $g(\cdot)$, the leader of the game which is the defender, solves the following optimization problem

$$J^*(K) = \min_K g(B^*(K)), \tag{4.16}$$

where J^* denotes the optimal value of J. In (4.16), $B^*(K)$ is the attacker's best response to the defender's strategy K. In fact, $B^*(K)$ is the solution to the following optimization problem

$$B^*(K) = \arg\max_B g(B). \tag{4.17}$$

Hence, the equilibrium strategy of the Stackelberg game leading to the defender's optimal strategy is given by

$$K^* = \arg\min_K J^*(K). \tag{4.18}$$

Algorithm 1 summarizes the procedure for the general case of multi attackers and multi defenders to find the solution of the Stackelberg game problem. Notably, the Stackelberg game problem might have non-unique solutions; however, the game-payoff is the same for all the solutions.

Algorithm 1 STACKELBERG ATTACKER–DEFENDER GAME IN VEHICLE PLATOONING

Input:
Data transfer structure (L_g), $g(\cdot)$, n, number of attacker(s) (f)
for $i = 1 : \binom{n}{f}$ **do** ▷ i is the defender index
  for $j = 1 : \binom{n}{f}$ **do** ▷ j is the attacker index
    $J(B(K_i)) = g(B_j)$
  end for
  $B^*(K_i) = \arg\max_{B_j} g(B_j)$
  $J(K_i) = g(B^*(K_i))$
end for
$J^*(K) = \min_{K_i} g(B^*(K_i))$
$K^* = \arg\min_K J^*(K)$
$B^* = \arg\max_B J(K^*)$
Output:
K^* ▷ Solution of the Stackelberg game problem (defender's optimal strategy)
B^* ▷ Solution of the Stackelberg game problem (attacker's optimal strategy)
$J(B^*, K^*)$ ▷ Optimal game pay-off

Referring back to our main objective, i.e., aiming at minimizing the aforementioned controllability metrics, we first need to calculate the controllability Gramian matrix associated with the attacked and defended closed-loop dynamics (4.13). The symmetric positive semidefinite controllability Gramian matrix associated with dynamics model (4.13) is given by

$$W_c(t) = \int_0^t e^{A_a s} B_a B_a^\mathsf{T} e^{A_a^\mathsf{T} s} ds, \tag{4.19}$$

which quantifies an energy-related measurement of the controllability level of the system. Eigenvectors of W_c corresponding to small eigenvalues determine directions in the state space that are less controllable (require large input energy to reach), and eigenvectors of W_c corresponding to large eigenvalues reflect directions in the state space that are more controllable (require small input energy to reach).

In case of an internally stable system, the state transition matrix $e^{A_a s}$ decays exponentially leading to the following finite positive definite controllability Gramian matrix

$$W_c = \int_0^\infty e^{A_a s} B_a B_a^\mathsf{T} e^{A_a^\mathsf{T} s} ds, \tag{4.20}$$

This infinite-horizon controllability Gramian matrix can also be computed by solving for the unique positive definite solution of the following Lyapunov equation

$$A_a W_c + W_c A_a^\mathsf{T} + B_a B_a^\mathsf{T} = \mathbf{0}. \tag{4.21}$$

Equation (4.21) forms a system of linear equations that can be easily solved. Dedicated algorithms have been proposed to solve for the solution of (4.21) effectively for even large scale systems [170–172]. In this chapter, we exploit the second method to find the infinite-horizon W_c in the interest of ease of computation. It is worth keeping in mind that since the closed-loop matrix A_a incorporates the unknown defender decision variable, K, derivation of a closed-form of W_c as a function of K is burdensome. Besides, we intend to focus on the optimal strategy of the defender for different information flow topologies rather than a general solution for W_c. Hence, we solve (4.21) for W_c numerically based on which we state the optimal defender strategy to mitigate the attack effects in different scenarios. It is noteworthy that our game formulation approach can be applied to increase the security level of platoons with an arbitrary number of followers equipped with different information flow topologies. Furthermore, other metrics could be exploited while applying our method for different assessments of the security-related aspects of a platoon.

4.3.4 Stability of the Closed-loop Platoon Dynamics

Generally, as a standard requirement, the stability of the closed-loop platoon dynamics needs to be ensured while performing the attack mitigation process. This requirement has to be met to have the desired rigid formation during our defined game. Basically, in platoon control, there are two different main stability notions defined for the closed-loop platoon dynamics, namely the *internal stability* and the *string stability*. A platoon with linear time-invariant dynamics is internally stable if and only if the least stable eigenvalue of the closed-loop system lies on the open left half-hand side of the complex plane [154]. For a platoon to be string stable, any disturbance introduced in the downstream of the platoon needs to be dampened while it is propagated along the upstream vehicles [165, 173]. In this chapter, we focus on the former and leave the latter for future work. Here, to have the vehicle platoon asymptotically stable prior to any attacks occur, we benefit from a result given in [68] to choose a set of controller gains to guarantee the asymptotically stability of

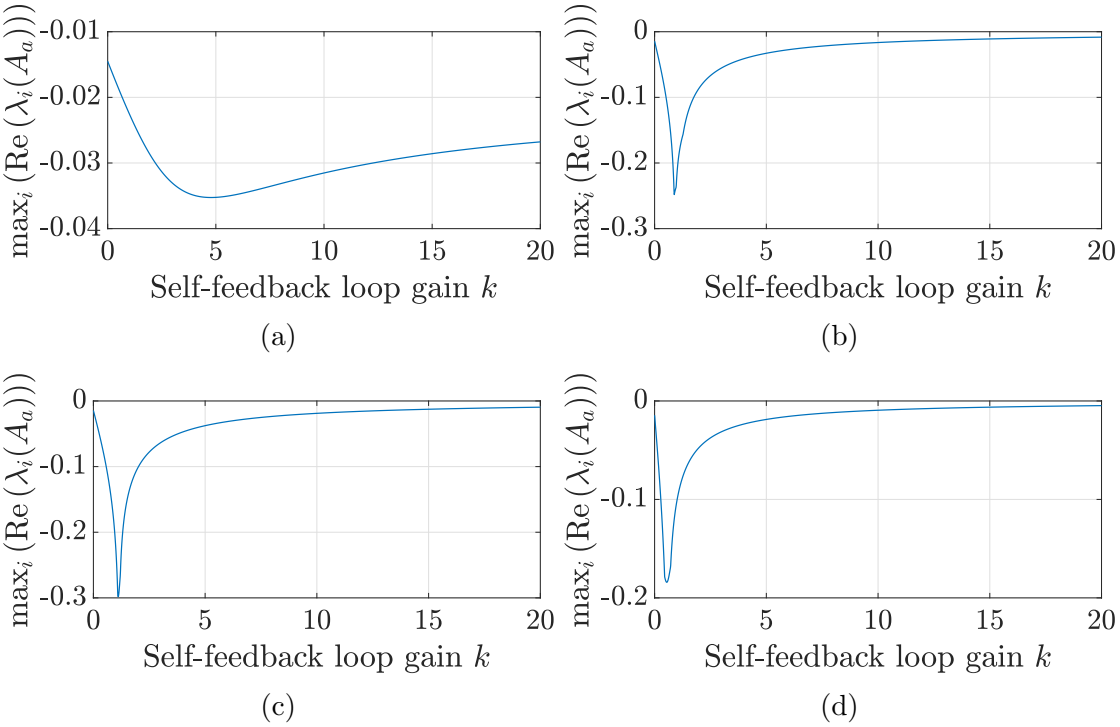


Figure 4.2: Closed-loop stability of platoon dynamics with different actuator placement and different self-loop gains (a): Actuator placed on vehicle 1, (b): Actuator placed on vehicle 6, (c): Actuators placed on vehicles 2 and 4, (d): Actuators placed on vehicles 5 and 6

the platoon under study as follows

$$\begin{cases} k_p > 0, \\ k_v > \dfrac{k_p \tau}{\min_{i\in\{1,2,\ldots,n\}}(\lambda_i k_a + 1)}, \\ k_a > -\dfrac{1}{\max_{i\in\{1,2,\ldots,n\}} \lambda_i}. \end{cases} \tag{4.22}$$

where λ_i is the eigenvalue of the grounded Laplacian matrix. Since we intend to focus on the optimal defender's strategy to mitigate the attack effects the most and the tuning of controller gains is not our concern, we simply choose $k_p = k_v = k_a = 1$, and $k = 2$ to satisfy the mentioned conditions and focus on the optimal defender's strategy for an asymptotically stable vehicle platoon. The value for the inertial lag is chosen as $\tau = 0.5$

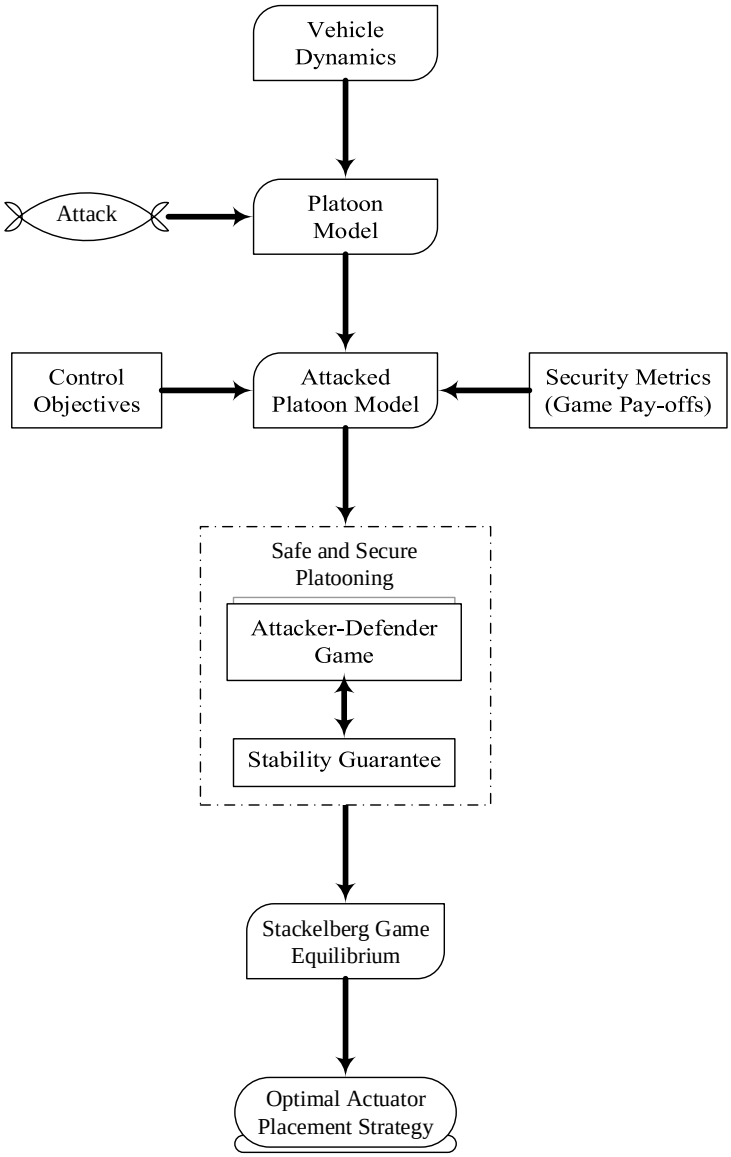


Figure 4.3: Proposed procedure to obtain the optimal strategy for the defender

sec throughout the following simulations. It is known that the inertial lag is bounded, i.e., $\tau \in [0, \tau_{\max}]$, where $\tau = 0$ corresponds to the ideal case of immediate actuation. Here, we choose a typical value reported in the literature [68]. Fig. 4.2 shows the stability

margin of the closed-loop system with the aforementioned values for k_p, k_v, and k_a for some different actuator placements with different gain values. It is obvious that in all of the scenarios, including the one we chose for our subsequent simulations ($k = 2$), the platoon is asymptotically stable. It turns out that the results hold for any set of controller gains as long as the platoon remains asymptotically stable.

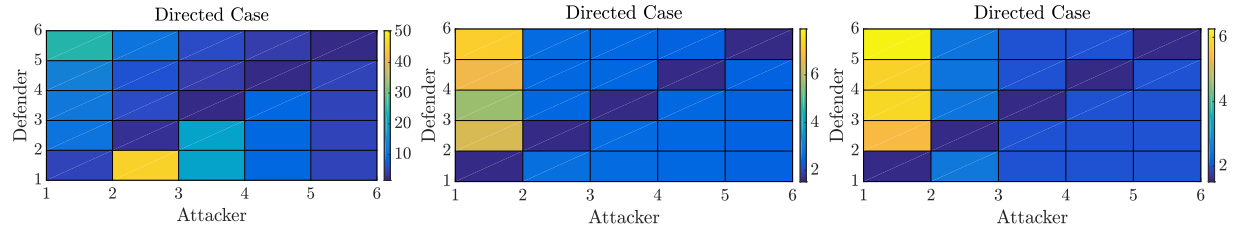
Fig. 4.3 shows the entire procedure to determine the optimal strategy for the defender.

4.4 Simulation Results

In this section, we present the simulation results stating the optimal actuator placement of the defender for a platoon with single attacker–single defender and multi attackers–multi defenders. Defender's optimal decision is determined based on the solution of the aforementioned Stackelberg game with the attacker(s) and the defender(s) as its players. Throughout the simulations, we consider a vehicle platoon consisting of 6 followers connected through the h–nearest neighbor information flow topology shown in Fig. 4.1.

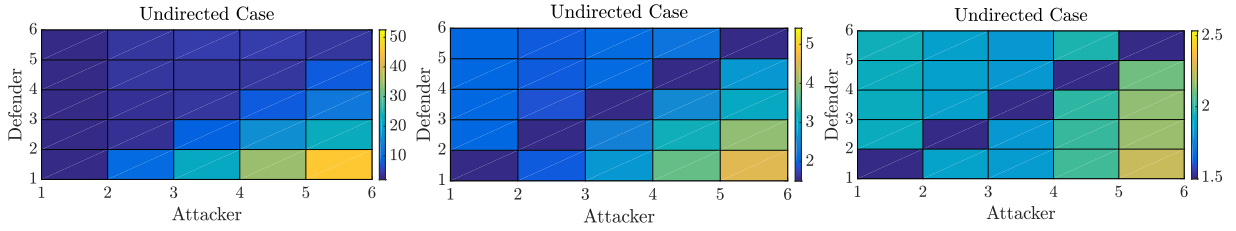
4.4.1 Single Attacker–Single Defender Platoon

Let us consider a vehicle platoon under cyber attack in which the vehicles can communicate with each other through unidirectional data transfer structure. We assume that each time one of the followers is attacked and for that particular case, the defender places his actuator on each of the followers. Fig. 4.4 and 4.6 illustrate the game pay-offs for all possible combinations of the attacker-defender for two different game pay-offs. One can easily see that increasing the connectivity among the vehicles cause the game values to be decreased. This, inherently, is a desired effect from the defender's perspective. As shown in these figures, both of the controllability metrics verify this result. Another significant point which needs to be highlighted is the fact that attacking the leader's neighbor in the unidirectional data transfer structure has the worst effect the attacker can impose on the platoon. In essence, the leader's neighbor is the vehicle receiving the most original version (with the least manipulations) of the reference profile generated by the leader. Basically, when vehicle 1 is attacked, it gets harder for the rest of the followers to receive the correct form of the reference profile. This is due to the critical role of the leader's neighbor who receives the intact reference profile from the leader and broadcasts to *all* of the followers regardless of the exploited information flow topology. This is the main reason that, for example, attacking vehicle 2 in a 2-nearest neighbor (or vehicle 3 in a 3-nearest neighbor)



(a) 1-nearest neighbor (b) 2-nearest neighbor (c) 3-nearest neighbor

Figure 4.4: Game pay-off for the attacked directed platoon with 6 followers and single attacker–single defender (game pay-off: $\lambda_{\max}(W_c)$)



(a) 1-nearest neighbor (b) 2-nearest neighbor (c) 3-nearest neighbor

Figure 4.5: Game pay-off for the attacked undirected platoon with 6 followers and single attacker–single defender (game pay-off: $\lambda_{\max}(W_c)$)

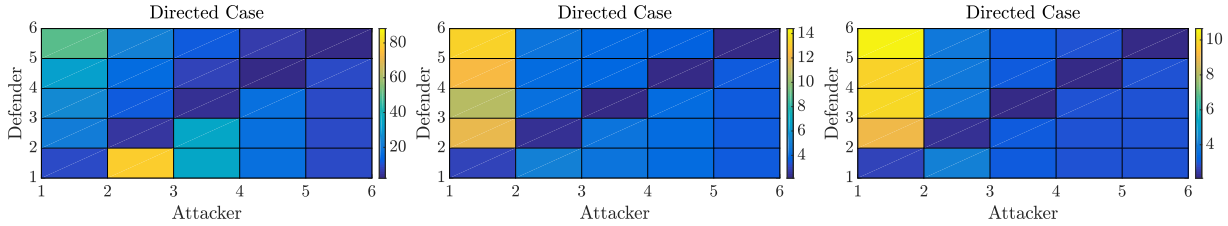
topology will not deteriorate the security level of the platoon as much as that of when attacking the leader's neighbor.

In the bidirectional information flow scenario, a similar attacked platoon is considered except the vehicles are able to send data over a bidirectional data transfer structure. Fig. 4.5 and 4.7 show the game pay-offs for this scenario. Note that a similar result regarding the benefit of increasing the connectivity of the platoon clearly holds for the bidirectional data transfer framework.

4.4.2 Multi Attackers–Multi Defenders Platoon

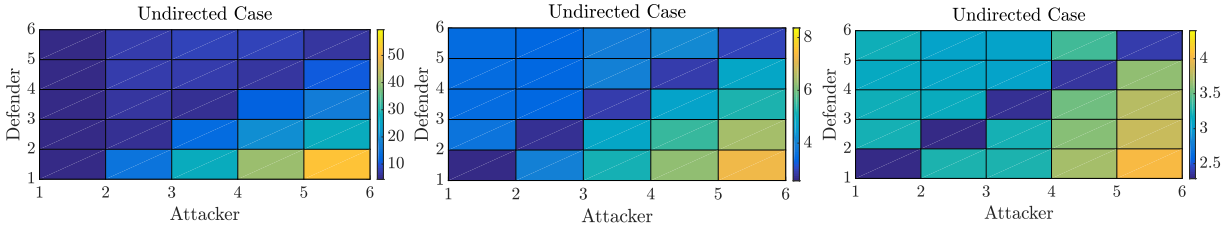
In this part, we assume the platoon is under cyber attacks imposed by more than one attacker. To avoid crowded figures in the simulations, we assume $f = 2$ and perform our analyses for both the unidirectional and bidirectional data transfer structures.

The same unidirectional and bidirectional platoons are considered except with two attackers and two defenders. The attackers might attack any pair of the vehicles and



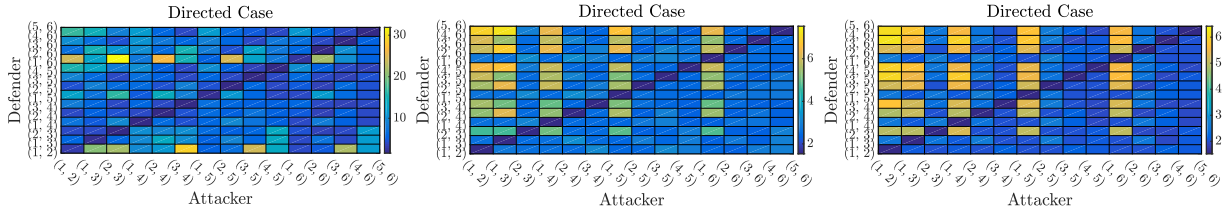
(a) 1-nearest neighbor (b) 2-nearest neighbor (c) 3-nearest neighbor

Figure 4.6: Game pay-off for the attacked directed platoon with 6 followers and single attacker–single defender (game pay-off: $\mathbf{tr}(W_c)$)



(a) 1-nearest neighbor (b) 2-nearest neighbor (c) 3-nearest neighbor

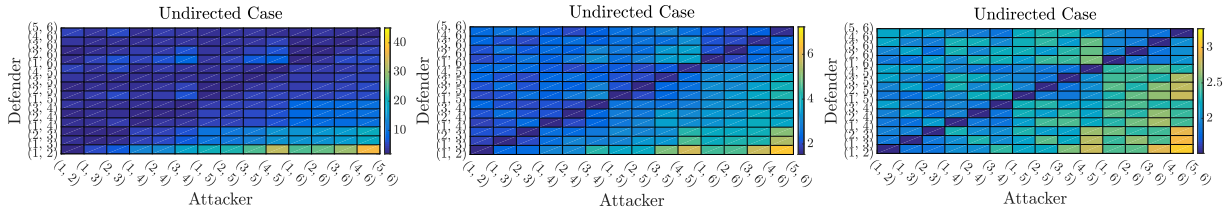
Figure 4.7: Game pay-off for the attacked undirected platoon with 6 followers and single attacker–single defender (game pay-off: $\mathbf{tr}(W_c)$)



(a) 1-nearest neighbor (b) 2-nearest neighbor (c) 3-nearest neighbor

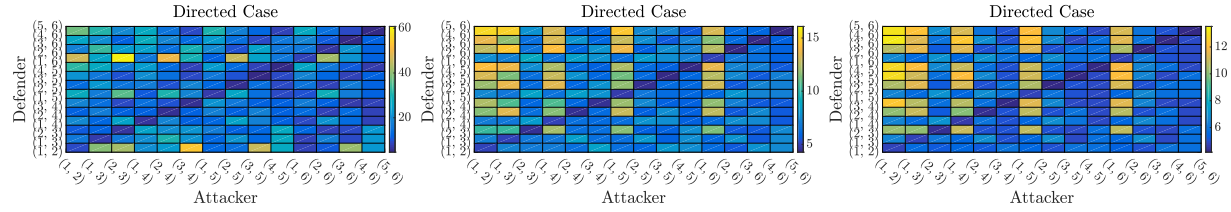
Figure 4.8: Game pay-off for the attacked directed platoon with 6 followers and two attackers–two defenders (game pay-off: $\lambda_{\max}(W_c)$)

based on the defender's strategy, the value of the game pay-off is calculated. Fig. 4.8 and 4.10 show the corresponding game values. As can be seen from these figures, a more densely connected platoon makes the energy needed by the attacker larger, hence, better from the defender's perspective. In this scenario, similar to the single attacker–single defender one, the attacker can effectively endanger the security of the platoon the most by including the leader's neighbor in his attacked vehicles.



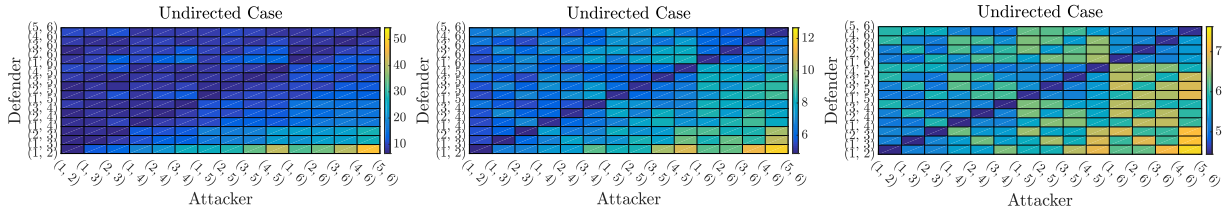
(a) 1-nearest neighbor (b) 2-nearest neighbor (c) 3-nearest neighbor

Figure 4.9: Game pay-off for the attacked undirected platoon with 6 followers and two attackers–two defenders (game pay-off: $\lambda_{\max}(W_c)$)



(a) 1-nearest neighbor (b) 2-nearest neighbor (c) 3-nearest neighbor

Figure 4.10: Game pay-off for the attacked directed platoon with 6 followers and two attackers–two defenders (game pay-off: $\mathbf{tr}(W_c)$)



(a) 1-nearest neighbor (b) 2-nearest neighbor (c) 3-nearest neighbor

Figure 4.11: Game pay-off for the attacked undirected platoon with 6 followers and two attackers–two defenders (game pay-off: $\mathbf{tr}(W_c)$)

From Fig. 4.9 and 4.11, it is again verified that as the connectivity among the vehicles is increased the attacker needs to exert more energy to perform the attack. Hence, the mentioned result holds regardless of the communication environment exploited in the platoon.

Remark 4.4. In all of the presented simulations, one can clearly see that, for any combination of the attacker(s) and the defender(s), the minimum game pay-off corresponds to the case where the defender exactly places its actuator(s) on the attacked node(s). Although

Table 4.1: Solution to the attacker–defender Stackelberg game (defender's optimal strategy) for the attacked platoons shown in Fig. 4.1 (the numbers represent the vehicle(s) on which the defender has to place his actuator(s))

	$f = 1$			
	Directed		**Undirected**	
	$\lambda_{\max}(W_c)$	$\mathbf{tr}(W_c)$	$\lambda_{\max}(W_c)$	$\mathbf{tr}(W_c)$
1-nearest neighbor	3	3	6	6
2-nearest neighbor	1	1	6	6
3-nearest neighbor	1	1	6	6
4-nearest neighbor	1	1	6	6
	$f = 2$			
	Directed		**Undirected**	
	$\lambda_{\max}(W_c)$	$\mathbf{tr}(W_c)$	$\lambda_{\max}(W_c)$	$\mathbf{tr}(W_c)$
1-nearest neighbor	(2, 4)	(2, 4)	(3, 6)	(3, 6)
2-nearest neighbor	(1, 4)	(1, 4)	(5, 6)	(5, 6)
3-nearest neighbor	(1, 2)	(1, 2)	(5, 6)	(5, 6)
4-nearest neighbor	(1, 2)	(1, 2)	(5, 6)	(5, 6)

this precise prediction may be unrealistic, it reflects the ideal decision that could be made by the defender. Furthermore, as the actuator(s) placement gets farther from the attacked nodes, the game pay-off increases, i.e., the attacker(s) needs to spend less energy to deviate the system towards his intended direction. □

Table 4.1 demonstrates the optimal defender's strategy for different information flow topologies, different game pay-offs, and different number of players for the considered platoon. The numbers represent the vehicle(s) on which the defender has to place his actuator(s). Inspired by this table, in a unidirectional vehicle platoon, it is more beneficial to place the actuator(s) at the downstream of the platoon. On the other hand, in a bidirectional vehicle platoon, placing the actuator(s) at the upstream of the platoon mitigates the attack effects more effectively. Similar results can be generated via the general method introduced in this chapter for any asymptotically stable platoon equipped with self-feedback loops.

Remark 4.5. Various energy-related controllability metrics might result in different control actions. In essence, in some instances, optimizing the trace of the controllability Gramian matrix can lead to a poor controllability performance in regard to the worst-case energy needed to reach a particular state; however, it is known that the selection of optimal nodes based on this metric benefits a closed-form solution as reported in the literature [161]. In such cases, if the analyses are performed by employing a numerical perspective without concerning about a closed-form solution, the largest eigenvalue of the controllability matrix is more suitable to be used as the game pay-off to ensure an appropriate controllability index of the system. □

Remark 4.6. It is remarkable that the simulation results show off-line game pay-offs for different combinations of the players. In fact, they are not the "final optimal" solution of the Stackelberg game problem which determines the optimal decisions that need to be made by the players. For instance, as was explained earlier, the game pay-offs demonstrate that (in a unidirectional topology) the attacker can endanger the security level of the platoon the most by targeting the leader's neighbor. This is exactly what the defender (as the game leader) figures out by solving the Stackelberg game problem to design his strategy. Hence, the defender eventually takes the optimal action accordingly to face this upcoming worst-case attack imposed by the intruder. □

4.5 Experimental Results

4.5.1 Basic Setup Architecture

As another verification approach to our results, we perform some experiments on a real platoon consisting of four scaled cars driving on a treadmill shown in Fig. 4.12. The positions, linear/angular velocities, steering, commanded throttle, actual throttle, and the State-of-Charge (SOC) of the batteries of each of the cars are exchanged between the host PC running the ROS and the vehicles through an IEEE 802.15.4-based 2.4GHz ZigBee wireless network protocol. Each of the cars is powered up by two identical batteries with the same initial SOC. The goal positions of the vehicles are commanded by the host PC, and the actual positions are captured via a central infrared camera detecting the specific Apriltags mounted on the vehicles. The vehicles find the position of their preceding car via the central camera and the PC (as the intermediate hardware) modeling the directed data transfer topology. Due to the limited length of the treadmill and to have the longest possible platoon, it is formed such that the four follower vehicles track a desired speed profile generated by the host PC which is considered as a virtual leader.

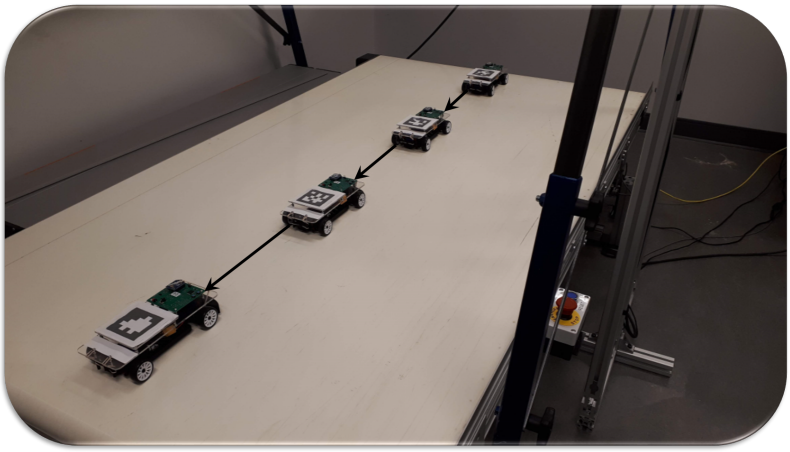


Figure 4.12: General schematic of the experimental platoon

4.5.2 Attack Mitigation Experiments

In our experiments, we consider the single attacker–single defender case. Following Algorithm 1 for a 1-nearest neighbor platoon composed of four homogeneous scaled cars with a directed data transfer structure and considering the largest eigenvalue of the controllability Gramian matrix as the game pay-off, it turns out that the optimal defender strategy is to place the self-loop on the second vehicle to mitigate the worst-case attack impact caused by attacking the third vehicle of the platoon. The controllability Gramian matrix highlighting the element which reflects the optimal decision made by the players is given by

$$\lambda_{\max}(W_c) = \begin{bmatrix} 1.5678 & 9.1645 & \boxed{5.2552} & 3.6413 \\ \boxed{4.3001} & 1.5605 & \boxed{5.2552} & \boxed{3.6413} \\ 6.0162 & 4.0937 & 1.5561 & 3.6413 \\ 10.0278 & 5.6221 & 3.8836 & 1.5504 \end{bmatrix} \tag{4.23}$$

Four separate experiments were conducted, each of which handled an acceleration attack on one of the individual cars along with implementing the defense mechanism on the second vehicle. In other words, in experiment i, the vehicle i is attacked, and the second vehicle defends. To reflect the amount of energy needed by the attacker to disrupt the desired rigid formation of the platoon, we consider the total dropped level of SOC of batteries of the attacked car.

Each experiment lasts for 10 minutes, wherein the first 2 minutes, we let the platoon simply run and reach its steady formation without any occurrence of attack or defense action. Having got a clear insight into the amount of dropped SOC values during our tests, we assume the attacker is able to impose his attacks with a periodic timing manner. This inherently models a severe attack scenario. It is worth keeping in mind that an intelligent attacker might not perform such a repetitive action in order not to get detected; however, in this study, we focus on the energy-related criterion of an attack rather than the attack detection. Hence, it is more appropriate for us to model such an intense attacker to have a clear view of the amount of energy he may need. The first attack occurs at $t = 2$ min., by injecting a sharp spike to the longitudinal position of the attacked car. The very first attack has a magnitude of 0.75 meters while the subsequent ones occur at every 2 seconds with a magnitude of 0.5 meters. The defense mechanism executed by vehicle 2 also has a periodic fashion. Particularly, it begins at time $t = 2$ min., lasting for 0.5 seconds with 3 seconds cooldown period between each defense. The total SOC values of the cars are sampled every 5 seconds and sent to the host PC via the XBee module. Fig. 4.13 shows the SOC values of the attacked car in each of the four experiments. As one can easily see from this figure, before the attack/defense onset, the SOCs decrease with a relatively low slope. Once the first attack occurs along with the defense of the second vehicle, the SOCs drop significantly. Thereafter, the SOCs decrease with relatively high slopes over the time course compared to the beginning of the experiments. As was expected, choosing the optimal solution of the Stackelberg game problem (in this case, the third vehicle), the intelligent attacker consumes the least amount of energy. Furthermore, based on the SOC values of the experiments, it is obvious that the intruder is never inclined to deviate his decision from this equilibrium point. It is also notable that experiment 2 verifies a trivial case. Rigorously, placement of both the actuator and the attack on the same node results in the most energy needed by the intruder (most dropped SOC value). However, this case is unlikely to happen in the real world. Let us consider the case in which the defender is the game leader (which is our focus in this work). In this case, the defender places its actuator on a specific node, followed by the attacker's decision. As the attacker is assumed to be aware of the defender's decision, he never attacks the defended node. On the other hand, let us consider the case where the attacker is the game leader. In this case, the intruder imposes his attack on a specific node, followed by the defender's decision. Thus, the defender definitely defends the exact same node attacked by the intruder resulting in maximizing the energy required by the attacker; however, in the real world the attacker's decision is not known beforehand.

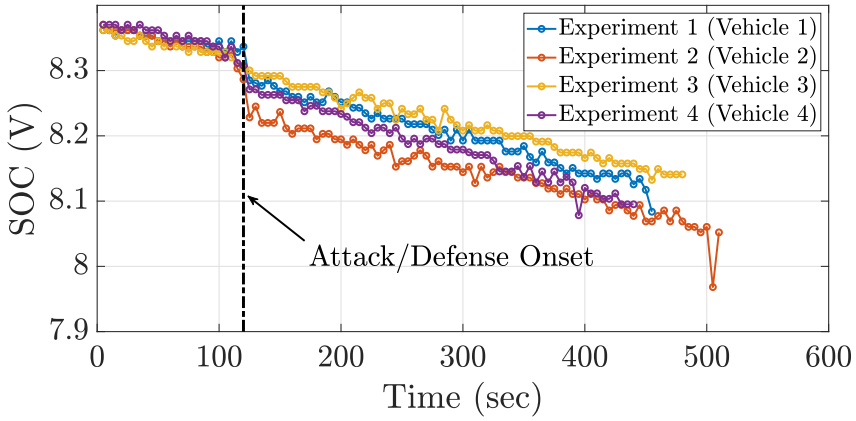


Figure 4.13: SOC values of the attacked car in each of the four experiments

4.6 Summary

In this chapter, a game-theoretic approach has been proposed to tackle the security challenges of vehicle platoons. From the viewpoint of secure platoon control, we studied the problem of threatening a vehicle platoon by one (or more) attacker(s) who tries to deteriorate the platoon control by injection of acceleration attack signal(s) to the longitudinal dynamics of one (or more) of the vehicles. In essence, we focused on the energy needed by the attacker, as our game-payoff, to steer the consensus dynamics of the system towards his desired direction in the state space. In this regard, the attacker(s) basically tries to minimize the amount of energy needed to deviate the system dynamics, while the defender(s) faces this action by attempting to maximize that energy. This confrontation between the attacker(s) and defender(s) was formulated as a Stackelberg game problem, and the algorithm to solve the game was given. Based on the equilibrium point of the game, the defender(s) selects specific nodes(s) to place his actuator(s) in order to mitigate the attack effects as far as possible. Two different scenarios, namely single attacker–single defender and multi attackers–multi defenders were considered. The game formulation, its solution, and simulation results were presented for h–nearest neighbor platoons with different data transfer structures. The proposed technique can be applied to arbitrary data transfer structures employed in different platoon formation topologies. Besides, the effects of increasing the connectivity among the vehicles on the security level of the platoon have been studied. Some experimental tests were also conducted on a real platoon to demonstrate the applicability of the method in practice. An open avenue for this research would be to generalize the work to study a platoon consisting of vehicles with different dynamics referred to as "heterogeneous" platoons.

Part II

Low Level Safe and Secure Vehicle Platoon Control

Chapter 5

Attack Resilient Nonlinear Heterogeneous Vehicle Platooning: Static Case

This chapter and the subsequent one deal with the secure control of vehicle platoons under the risk of a common cyber attack, namely Denial-of-Service (DoS) attack. A malicious DoS intruder can endanger the security of platoon by jamming the communication network among the vehicles which is responsible to transmit inter-vehicular data throughout the platoon. In other words, he may cause a failure in the network by jamming it or injecting a huge amount of delay, which in essence makes the outdated transferred data useless. In fact, a DoS attacker uses the disruption resources to violate data integrity or availability, hence, causes a blockage or at least suffering delays in data transfer in the network by making the beacon nodes unnecessarily busy. This can potentially result in huge performance degradation or even hazardous collisions. Researchers have devoted efforts to address the DoS attack in networked control systems and platoons; however, they mostly limit the study to linear models along with a linear controller applied to homogeneous platoons and assume perfect communication links [61, 174–176]. We consider a general heterogeneous platoon under DoS attack with nonlinear vehicle dynamics in our analyses. It is shown that our proposed algorithm mitigates the effect of the attack while the desired platoon formation is maintained. Besides, our proposed method can handle different communication topologies, which are some other missing contributions in the literature. We propose a secure distributed nonlinear model predictive control algorithm consisting of i) detection and ii) mitigation phases. The algorithm is capable of handling DoS attack performed on a platoon equipped by different communication topologies and at the same

time it guarantees the desired formation control performance. Stability analysis of the attacked platoon running the given algorithm is also presented. Simulation results on a sample heterogeneous attacked platoon exploiting two-predecessor follower communication environment demonstrates the effectiveness of the method.

The results of this chapter have been published in [177].

5.1 Organization of the Chapter

The remainder of this chapter is organized as follows. Sec. 5.2 defines the problem statement, including the control objectives and the attack model. Sec. 5.3 details the main results and describes the proposed algorithm. Stability analysis of the attacked platoon employing the proposed method is presented in the section. Simulation results on a sample heterogeneous attacked platoon demonstrating the fruitfulness of the method are given in Sec. 5.4. Finally, Sec. 5.5 concludes the chapter and gives some open avenues to continue this work.

5.2 Problem Statement

5.2.1 System Model

We consider a heterogeneous platoon consisting of n follower vehicles each of which modeled with the following discrete-time nonlinear dynamics model [61]

$$\begin{cases} p_i(t+1) = p_i(t) + v_i(t)\Delta t, \\ v_i(t+1) = v_i(t) + \dfrac{\Delta t}{M_i}\left(\eta_{T,i}\dfrac{T_i(t)}{R_{w,i}} - C_{A,i}v_i^2(t) - M_i g f_{r,i}\right) \\ T_i(t+1) = T_i(t) - \dfrac{1}{\tau_i}T_i(t)\Delta t + \dfrac{1}{\tau_i}u_i\Delta t, \;\; i = 1, 2, \ldots, n \end{cases} \tag{5.1}$$

where Δt is the sampling time, $p_i(t)$ and $v_i(t)$ are the position and velocity of vehicle i, respectively. M_i denotes the mass, $C_{A,i}$ is the coefficient of aerodynamic drag, $f_{r,i}$ is the coefficient of rolling resistance, g is the gravity constant, $T_i(t)$ is the actual driving/braking torque applied to the drivetrain, $R_{w,i}$ is the tire radius, τ_i is the inertial lag of vehicle powertrain, $\eta_{T,i}$ is the mechanical efficiency of the driveline, and $u_i(t)$ is the desired driving/braking torque which represents the control input.

The states and outputs of each vehicle are represented by $\boldsymbol{x}_i(t) = [p_i(t), v_i(t), T_i(t)]^\mathsf{T}$, and $\boldsymbol{y}_i(t) = [p_i(t), v_i(t)]^\mathsf{T}$, respectively. Collecting the nonlinear terms of the dynamics (5.1) creates a more compact form

$$\begin{cases} \boldsymbol{x}_i(t+1) = \boldsymbol{\phi}_i\left(\boldsymbol{x}_i(t)\right) + \boldsymbol{\psi}_i u_i(t) \\ \quad \boldsymbol{y}_i(t) = \boldsymbol{\gamma}\boldsymbol{x}_i(t). \end{cases} \tag{5.2}$$

where $\boldsymbol{\psi}_i = [0, 0, (1/\tau_i)\Delta t]^\mathsf{T}$, $\boldsymbol{\gamma} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$, and

$$\boldsymbol{\phi}_i = \begin{bmatrix} p_i(t) + v_i(t)\Delta t \\ v_i(t) + \dfrac{\Delta t}{M_i}\left(\eta_{T,i}\dfrac{T_i(t)}{R_{w,i}} - C_{A,i}v_i^2(t) - M_i g f_{r,i}\right) \\ T_i(t) - (1/\tau_i)T_i(t)\Delta t \end{bmatrix}$$

Stacking the states, outputs, and the control input signals of all vehicles into vectors yields the platoon dynamics as follows

$$\begin{cases} X(t+1) = \boldsymbol{\Phi}(X(t)) + \boldsymbol{\Psi}U(t), \\ Y(t+1) = \boldsymbol{\Theta} \cdot X(t+1), \end{cases} \tag{5.3}$$

where $X(t) = [x_1(t)^\mathsf{T}, x_2(t)^\mathsf{T}, \ldots, x_n(t)^\mathsf{T}]^\mathsf{T} \in \mathbb{R}^{3n\times 1}$, $Y(t) = [y_1(t)^\mathsf{T}, y_2(t)^\mathsf{T}, \ldots, y_n(t)^\mathsf{T}]^\mathsf{T} \in \mathbb{R}^{2n\times 1}$, $U(t) = [u_1(t), u_2(t), \ldots, u_n(t)]^\mathsf{T} \in \mathbb{R}^{n\times 1}$. Besides, $\boldsymbol{\Phi} = [\phi_1^\mathsf{T}, \phi_2^\mathsf{T}, \ldots, \phi_n^\mathsf{T}]^\mathsf{T} \in \mathbb{R}^{3n\times 1}$, $\boldsymbol{\Psi} = \text{diag}\{\psi_1, \psi_2, \ldots, \psi_n\} \in \mathbb{R}^{3n\times n}$, and $\boldsymbol{\Theta} = I_N \otimes \boldsymbol{\gamma} \in \mathbb{R}^{2n\times 3n}$.

We model the communication links among the vehicles in the platoon by a graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ where $\mathcal{V}$ and $\mathcal{E}$ denote the set of nodes (vehicles) and the edges (modeling the links between the vehicles), respectively [178]. Followed by this definition, the adjacency, in-degree, and pinning matrices are defined as

$$\boldsymbol{\mathcal{A}} = [a_{ij}] = \begin{cases} a_{ij} = 1, & \text{if } \{j, i\} \in \mathcal{E} \\ a_{ij} = 0, & \text{if } \{j, i\} \notin \mathcal{E} \end{cases} \tag{5.4}$$

$$\boldsymbol{\mathcal{D}} = \text{diag}\{\deg_1, \deg_2, \ldots, \deg_n\} \tag{5.5}$$

$$\boldsymbol{\mathcal{P}} = \text{diag}\{p_1, p_2, \ldots, p_n\} \tag{5.6}$$

where $\deg_i = \Sigma_{j=1}^n a_{ij}$, and $p_i = 1$ if the leader vehicle can send data to vehicle i and $p_i = 0$ otherwise. Furthermore, we define the neighbor set of vehicle i as $\mathbb{N}_i = \{j \mid a_{ij} = 1, j =$

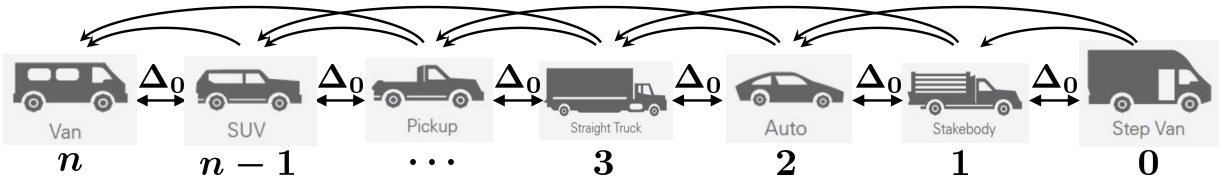


Figure 5.1: TPF heterogeneous platoon consisted of n followers

$1, 2, \ldots, n\}$ which are the vehicles that can send data to vehicle i. Besides, we define the set $\mathbb{O}_i = \{j \mid a_{ji} = 1, j = 1, 2, \ldots, n\}$ which are the vehicles that can receive data from vehicle i. If vehicle i can also receive data from the leader, then $\mathbb{I}_i = \mathbb{N}_i \cup \{0\}$, otherwise $\mathbb{I}_i = \mathbb{N}_i$. In this chapter, for convenience, we consider a heterogeneous platoon equipped by Two-Predecessor Follower (TPF) communication topology as an example shown in Fig. 5.1; however, it is easy to adapt our algorithm for other communication topologies.

5.2.2 Platoon Control Objectives

The platoon control objective is for the followers to track the reference speed profile generated by the leader while maintaining a constant distance between any two consecutive vehicles, i.e.

$$\begin{cases} \lim\limits_{t\to\infty} \|v_i(t) - v_0(t)\| = 0, \\ \lim\limits_{t\to\infty} \|p_{i-1}(t) - p_i(t) - \Delta_{i-1,i}\| = 0, \end{cases} \quad i = 1, 2, \ldots, n, \tag{5.7}$$

where $\Delta_{i-1,i} = \Delta_0$ is the desired constant space between consecutive vehicles $i-1$ and i.

5.2.3 Attack Description

We focus on a widely spread cyber attack, called the DoS attack. Basically, a DoS attacker jeopardizes the security of the system through jamming the network by flooding it with fake requests so that the shared network becomes overwhelmed by these demands, hence, is too busy to process the legitimate requests sent by the authorized users [105, 179]. This inherently causes packet loss or at least suffering delays in data transfers. In our application, we assume that the DoS attacker is able to block the communication link among two nonconsecutive neighboring vehicles which results in missing inter-vehicular data received by the follower vehicle.[1] In essence, if the communication link among vehicle

[1] In the next chapter, we will extend our results to another DoS attack modeling presented in the literature.

i and $i-2$ is attacked during $t \in [t_0, t_1]$, the vehicle i is only able to receive the valid data up to $t = t_0$ and has the exact same data until the attack is over, i.e., vehicle i will restart to receive updated data from vehicle $i-2$ at $t > t_1$. In the rest of the chapter, we denote $\tau_a = t_1 - t_0$ as the attack period for notational convenience.

Assumption 5.1. As a standard assumption and from a practical point of view, we assume that the attacker has a limited amount of energy resources that prevents him from jamming the network ceaselessly [111, 180, 181]. □

Remark 5.2. It is remarkable that the DoS attacker never attacks a link among consecutive vehicles. This is due to fact that in our algorithm the positions and velocities are transmitted which can be reliably measured by on-board sensors mounted on an ego-vehicle such as GPS and radar. Hence, once a follower detects that those quantities are no longer updated, it can switch to its redundant sensors to have real time data. □

5.3 Main Results

5.3.1 Secure–DNMPC for Vehicle Platooning

To combat the DoS attacker described in the previous section, we exploit a modified version of the Distributed Nonlinear Model Predictive Control (DNMPC) approach proposed in [61], called Secure–DNMPC which aims at mitigating the effects of the attack while achieving the desired control objectives. The algorithm is basically composed of two main phases, namely i) the detection and ii) the mitigation phase. In the first phase, we attempt to detect if a DoS attack is underway. If an attack is detected such that the communication link connecting the ego-vehicle with its immediate preceding or following vehicle is endangered, then the algorithm commands the victim vehicle to ignore the data received through the V2V link (until the attack is over) and switch to its on-board sensors followed by the implementation of the DNMPC. Otherwise, if the blocked link corresponds to the farther neighbors of the ego-vehicle, the victim vehicle makes use of the most recent updated data prior to the attack commence and the mitigation phase starts by performing Secure–DNMPC. Inherently, in the second phase, each vehicle solves a local optimal control problem given as follows to generate its own optimal control input signal which then needs to be exchanged with its neighbors.

Problem 5.3 (Local NMPC)**.** Each vehicle i has to solve a local NMPC problem at each time instant t to get its own optimal control input and exchange it with its neighbors as follows

Local NMPC: Vehicle $i = 1, 2, \ldots, n$, has to solve the following optimization problem with prediction horizon N_p at each time instant t

$$\begin{aligned}
&\min_{u_i^p(0|t-\tau_a),\ldots,u_i^p(N_p-1|t-\tau_a)} J_i(\boldsymbol{y}_i^p, u_i^p, \boldsymbol{y}_i^a, \boldsymbol{y}_{-i}^a) \\
&\quad = \Sigma_{k=0}^{N_p-1} \Big(\left\| \boldsymbol{y}_i^p(k|t-\tau_a) - \boldsymbol{y}_{\mathrm{des},i}(k|t-\tau_a) \right\|_{\boldsymbol{Q}_i} \\
&\quad + \left\| u_i^p(k|t-\tau_a) - h_i\left(v_i^p(k|t-\tau_a)\right) \right\|_{R_i} \\
&\quad + \left\| \boldsymbol{y}_i^p(k|t-\tau_a) - \boldsymbol{y}_i^a(k|t-\tau_a) \right\|_{\boldsymbol{F}_i} \\
&\quad + \Sigma_{j\in\mathbb{N}_i} \left\| \boldsymbol{y}_i^p(k|t-\tau_a) - \boldsymbol{y}_{-j}^a(k|t-\tau_a) - \tilde{\Delta}_{i,j} \right\|_{\boldsymbol{G}_i} \Big)
\end{aligned} \tag{5.8a}$$

Subject to:

$$\boldsymbol{x}_i^p(k+1|t-\tau_a) = \boldsymbol{\phi}\left(\boldsymbol{x}_i^p(k|t-\tau_a)\right) + \boldsymbol{\psi} u_i^p(k|t-\tau_a) \tag{5.8b}$$

$$\boldsymbol{y}_i^p(k|t-\tau_a) = \boldsymbol{\gamma} \boldsymbol{x}_i^p(k|t-\tau_a) \tag{5.8c}$$

$$\boldsymbol{x}_i^p(0|t-\tau_a) = \boldsymbol{x}_i(t-\tau_a) \tag{5.8d}$$

$$u_i^p(k|t-\tau_a) \in \mathfrak{U}_i \tag{5.8e}$$

$$\boldsymbol{y}_i^p(N_p|t-\tau_a) = \frac{1}{|\mathbb{I}_i|} \Sigma_{j\in\mathbb{I}_i} \left(\boldsymbol{y}_{-j}^a(N_p|t-\tau_a) + \tilde{\Delta}_{i,j} \right) \tag{5.8f}$$

$$T_i^p(N_p|t-\tau_a) = h_i\left(v_i^p(N_p|t-\tau_a)\right) \tag{5.8g}$$

where $\boldsymbol{y}_{-i}(t) := [\boldsymbol{y}_{i_1}^\top, \ldots, \boldsymbol{y}_{i_m}^\top]^\top$ (if $\{i_1, \ldots, i_m\} := \mathbb{N}_i$), $\boldsymbol{y}_{\mathrm{des},i}(t) = \boldsymbol{\gamma} \boldsymbol{x}_{\mathrm{des},i}(t)$, $\boldsymbol{x}_{\mathrm{des},i}(t) = [p_0(t) - i\Delta_0, v_0, h_i(v_0)]^\mathsf{T}$, $h_i(v_0) = \frac{R_{w,i}}{\eta_{T,i}} \times (C_{A,i} v_0^2 + M_i g f_{r,i})$, $\mathfrak{U}_i = \{u_i \mid u_i \in [\underline{u}_i, \bar{u}_i]\}$ defines the feasible bounds on the control input, $\tilde{\Delta}_{i,j} = [\Delta_{i,j}, 0]^\mathsf{T}$ denotes the desired spacing between the vehicles i and j, and $\boldsymbol{Q}_i$, R_i, $\boldsymbol{F}_i$, and $\boldsymbol{G}_i$ are the NMPC tuning weight matrices. All the aforementioned weighting matrices are assumed to be symmetric and satisfy the following conditions [61]:

1. $\boldsymbol{Q}_i \succeq 0$, which is to penalize the deviation of the output from the desired equilibrium. It is notable that $\boldsymbol{Q}_i$ also determines whether node i is pinned to the leader. If $p_i = 0$, then node i is unable to know its desired set point, and therefore, $\boldsymbol{Q}_i = 0$ is always enforced. If $p_i = 1$, then $\boldsymbol{Q}_i \succ 0$ in its penalization functions.

2. $R_i \succeq 0$, which represents the strength to penalize the input error diverged from equilibrium, meaning that the controller prefers to keep constant speed.

3. $\boldsymbol{F}_i \succeq 0$, which means that node i tries to maintain its assumed output. Notably, this assumed output is actually the shifted last-step optimal trajectory of the same node, and this output is sent to the neighbor nodes in set $\mathbb{O}_i$.

4. $\boldsymbol{G}_i \succeq 0$, which states that node i tries to maintain the output as close to the assumed trajectories of its neighbors (i.e., $j \in \mathbb{N}_i$) as possible.

$\boldsymbol{y}_i^a(t)$ represents the data sent by the vehicle i to the set $\mathbb{O}_i$ while $\boldsymbol{y}_{-j}^a$ denotes the data received by the vehicle i from its neighbors $j \in \mathbb{N}_i$. The penultimate constraint referred to as the terminal averaging constraint is to enforce the vehicle i to have the same output as the average of assumed outputs in $\mathbb{I}_i$ at the end of the prediction horizon. The last terminal constraint is to enforce vehicle i to drive with a constant speed at the end of the prediction horizon. These two constraints are necessary for the stability of the DMPC algorithm [61]. Superscript a, p, and $*$ are to distinguish between assumed, predicted and optimal quantities, respectively. The assumed quantities are the ones transmitted by the vehicles in the platoon.

Fig. 5.2 shows a flowchart illustrating the procedure of the Secure–DNMPC design.

Based on the above problem, the proposed approach is described in detail in Algorithm 2.

5.3.2 Stability Analysis of Secure–DNMPC

In this section we analyze the stability of the Secure–DNMPC algorithm which incorporates the time delay τ_a imposed by the DoS attacker. Prior to stability analysis, we first introduce the following Lemmas.

Lemma 5.4 ([182]). *The eigenvalues of Kronecker product of two matrices $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{m\times m}$ are*

$$\lambda_i \mu_j, \quad i = 1, 2, \ldots, n, \quad j = 1, 2, \ldots, m,$$

where λ_i and μ_j are the eigenvalues of A and B, respectively. □

Lemma 5.5 ([61]). *For any platoon wherein all the vehicles can receive data (directly/indirectly) from the leader vehicle, the eigenvalues of $(\boldsymbol{\mathcal{D}} + \boldsymbol{\mathcal{P}})^{-1} \boldsymbol{\mathcal{A}}$ lie within the unit circle disk, i.e.*

$$\left|\lambda_i \left\{(\boldsymbol{\mathcal{D}} + \boldsymbol{\mathcal{P}})^{-1} \boldsymbol{\mathcal{A}}\right\}\right| < 1. \tag{5.9}$$

□

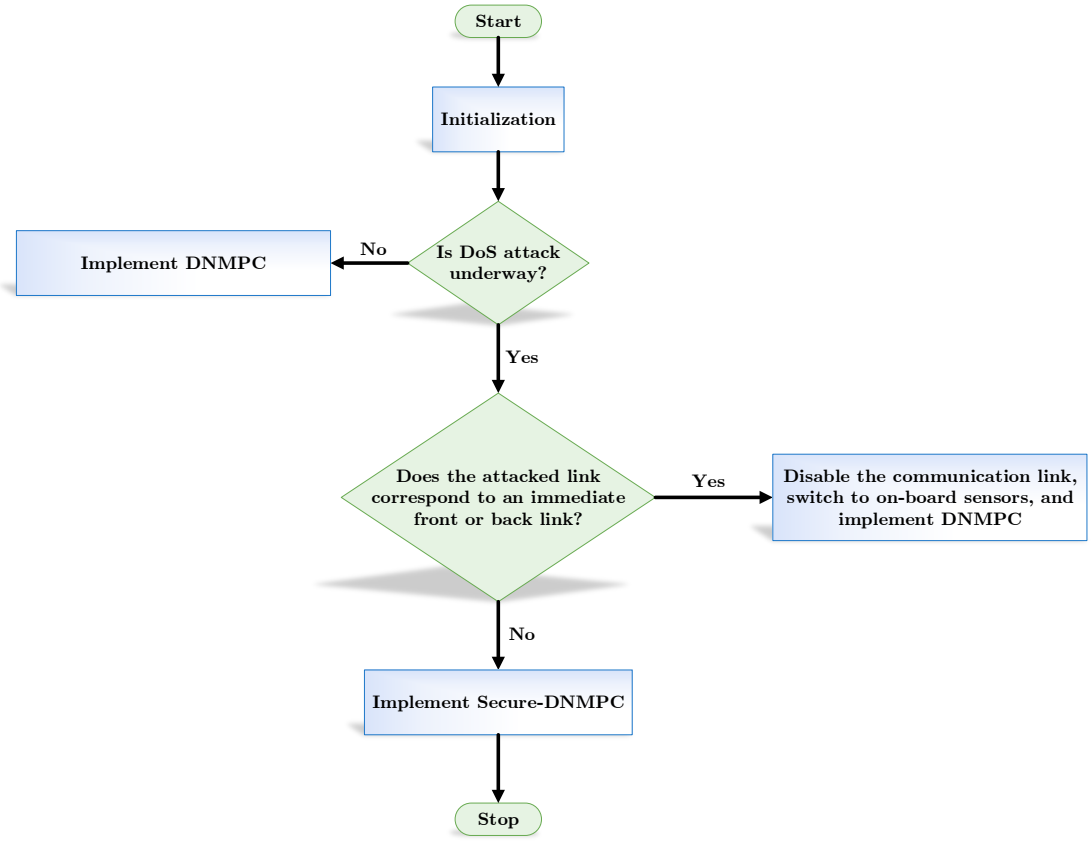


Figure 5.2: Procedure of the proposed Secure–DNMPC design

Now, we can prove the stability of the Secure–DNMPC algorithm.

Theorem 5.6. *If a platoon which is under a DoS attack satisfies the condition in Lemma 5.5, then the terminal output of the system controlled by the Secure–DNMPC proposed in Algorithm 2 asymptotically converges to the desired state, i.e.*

$$\lim_{t\to\infty} \left| \boldsymbol{y}_i^p(N_p|t-\tau_a) - \boldsymbol{y}_{des,i}(N_p|t-\tau_a) \right| = 0. \tag{5.10}$$

□

Proof: First of all, we state that a suitable Lyapunov candidate to prove the asymptotic stability is the sum of all local cost functions introduced in the local NMPC problem as suggested by [183]

$$J_\Sigma^*(t-\tau_a) = \Sigma_{i=1}^n J_i^* \left(\boldsymbol{y}_i^*(:|t-\tau_a), u_i^*(:|t-\tau_a), \boldsymbol{y}_i^a(:|t-\tau_a), \boldsymbol{y}_{-i}^a(:|t-\tau_a) \right).$$

Algorithm 2 SECURE–DNMPC FOR VEHICLE PLATOONING UNDER DOS ATTACK

1: **Initialization:**
Assumed values for vehicle i are set at time $t = 0$,
$u_i^a(k|0) = h_i(v_i(0)), \boldsymbol{y}_i^a(k|0) = \boldsymbol{y}_i^p(k|0), \quad k = 0, 1, \ldots, N_p - 1$
2: **while** $t \leq t_{\text{final}}$ **do**
3: **if** $p_{-j}^a(t) = p_{-j}^a(t-1), j \in \mathbb{N}_i$ **then** ▷ Check to see if a DoS is underway
4: **if** $j = i - 1$ or $j = i + 1$ **then** ▷ Check to see if the attacked link
5: corresponds to a predecessor or a follower
6: Disable communication link, switch to on-board sensors & **Go to:** 8
7: **else**
8: **for** Each vehicle i **do** ▷ Implement Secure–DNMPC
9: Solve Problem 5.3 at time $t > 0$ and yield $u_i^*(k|t-\tau_a), \quad k = 0, 1, \ldots, N_p - 1$
10: Compute: $\begin{cases} \boldsymbol{x}_i^*(k+1|t-\tau_a) = \boldsymbol{\phi}_i(\boldsymbol{x}_i^*(k|t-\tau_a)) + \boldsymbol{\psi}_i u_i^*(k|t-\tau_a), \\ \boldsymbol{x}_i^*(0|t-\tau_a) = \boldsymbol{x}_i(t-\tau_a), \quad k = 0, 1, \ldots, N_p - 1 \end{cases}$
11: Compute: $u_i^a(k|t-\tau_a+1) = \begin{cases} u_i^*(k+1|t-\tau_a), & k = 0, 1, \ldots, N_p - 2 \\ h_i\left(v_i^*(N_p|t-\tau_a)\right), & k = N_p - 1 \end{cases}$
12: Compute: $\begin{cases} \boldsymbol{x}_i^a(k+1|t-\tau_a+1) = \boldsymbol{\phi}_i\left(\boldsymbol{x}_i^a(k|t-\tau_a+1)\right) + \boldsymbol{\psi}_i u_i^a(k|t-\tau_a+1) \\ \boldsymbol{x}_i^a(0|t-\tau_a+1) = \boldsymbol{x}_i^*(1|t-\tau_a), \quad k = 0, 1, \ldots, N_p - 1 \end{cases}$
13: Compute: $\boldsymbol{y}_i^a(k|t-\tau_a+1) = \boldsymbol{\gamma}\boldsymbol{x}_i^a(k|t-\tau_a+1), \quad k = 0, 1, \ldots, N_p - 1$
14: Send $\boldsymbol{y}_i^a(k|t-\tau_a+1)$ to the vehicles lie in the set $\mathbb{O}_i$, and receive $\boldsymbol{y}_{-j}^a(k|t-\tau_a+1)$ from neighboring vehicles $j \in \mathbb{N}_i$ and compute $\boldsymbol{y}_{\text{des},i}(k|t-\tau_a+1)$
15: Exert the first element of the optimal control signal $u_i(t-\tau_a) = u_i^*(0|t-\tau_a)$
16: **end for**
17: **end if**
18: **end if**
19: **end while**

Inspired by a similar idea used in [61], we define the tracking error output vector

$$\tilde{\boldsymbol{y}}_i^p(N_p|t-\tau_a) = \boldsymbol{y}_i^p(N_p|t-\tau_a) - \boldsymbol{y}_{\text{des},i}(N_p|t-\tau_a) \tag{5.11}$$

As we assumed that the followers have zero acceleration at the end of the horizon, using the update control law defined in line 11 of the Algorithm 2 we get

$$\boldsymbol{y}_i^a(N_p|t-\tau_a+1) = \boldsymbol{y}_i^p(N_p|t-\tau_a) + E\boldsymbol{y}_i^p(N_p|t-\tau_a)\Delta t, \tag{5.12}$$

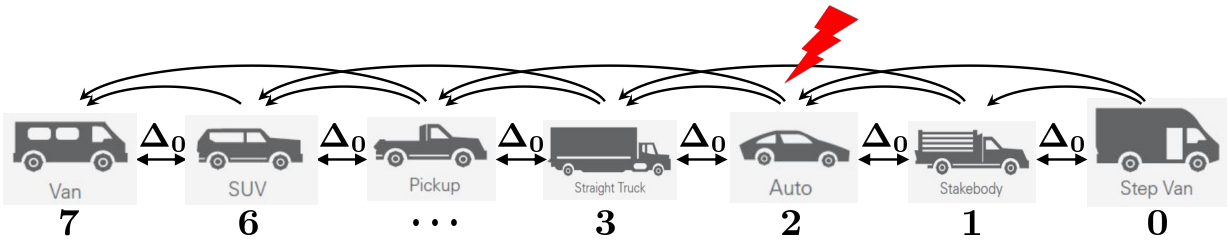


Figure 5.3: TPF heterogeneous platoon imposed by a DoS attacker on the communication link between vehicle 1 and 3

where $E = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$. Combining (5.12) with the terminal averaging constraint together with considering the defined tracking error vector in (5.11) yields

$$\tilde{\boldsymbol{y}}_i^p(N_p|t-\tau_a+1) = \frac{1}{|\mathbb{I}_i|}\Sigma_{j\in\mathbb{I}_i}(I_2 + E\Delta t)\tilde{\boldsymbol{y}}_j^p(N_p|t-\tau_a). \tag{5.13}$$

This can be rewritten in a matrix format

$$\tilde{Y}^p(N_p|t-\tau_a+1) = \left((\boldsymbol{\mathcal{D}}+\boldsymbol{\mathcal{P}})^{-1}\boldsymbol{\mathcal{A}}\right) \otimes (I_2 + E\Delta t)\tilde{Y}^p(N_p|t-\tau_a) \tag{5.14}$$

where $\tilde{Y}^p(N_p|t-\tau_a) = [\tilde{\boldsymbol{y}}_j^p(N_p|t-\tau_a), \ldots, \tilde{\boldsymbol{y}}_j^p(N_p|t-\tau_a)]^\mathsf{T} \in \mathbb{R}^{2n\times 1}$. Based on Lemma 5.5 the eigenvalues of $(\boldsymbol{\mathcal{D}}+\boldsymbol{\mathcal{P}})^{-1}\boldsymbol{\mathcal{A}}$ lie within the unit circle disk. Besides, one can easily check that the eigenvalues of $I_2 + E\Delta t$ are all equal to one. Hence, according to Lemma 5.4, the eigenvalues of $((\boldsymbol{\mathcal{D}}+\boldsymbol{\mathcal{P}})^{-1}\boldsymbol{\mathcal{A}}) \otimes (I_2 + E\Delta t)$ lie within the unit circle disk. This together with (5.14) completes the proof. ■

Remark 5.7. It is notable that in case of emergencies caused by a severe attacker, provided that the energy constraint is of lower priorities, one can reduce R_i in the cost function defined in the local NMPC problem to let a more mitigation level against the intruder. In the extreme case, $R_i = 0$ lets the controller exert as much control effort as possible to have a safe and secure platooning. □

5.4 Simulation Results

A heterogeneous platoon consisted of seven different followers is considered where they can exchange inter-vehicular data among each other through the TPF communication topology. It is assumed that the communication link among the vehicle 1 and 3 is subject

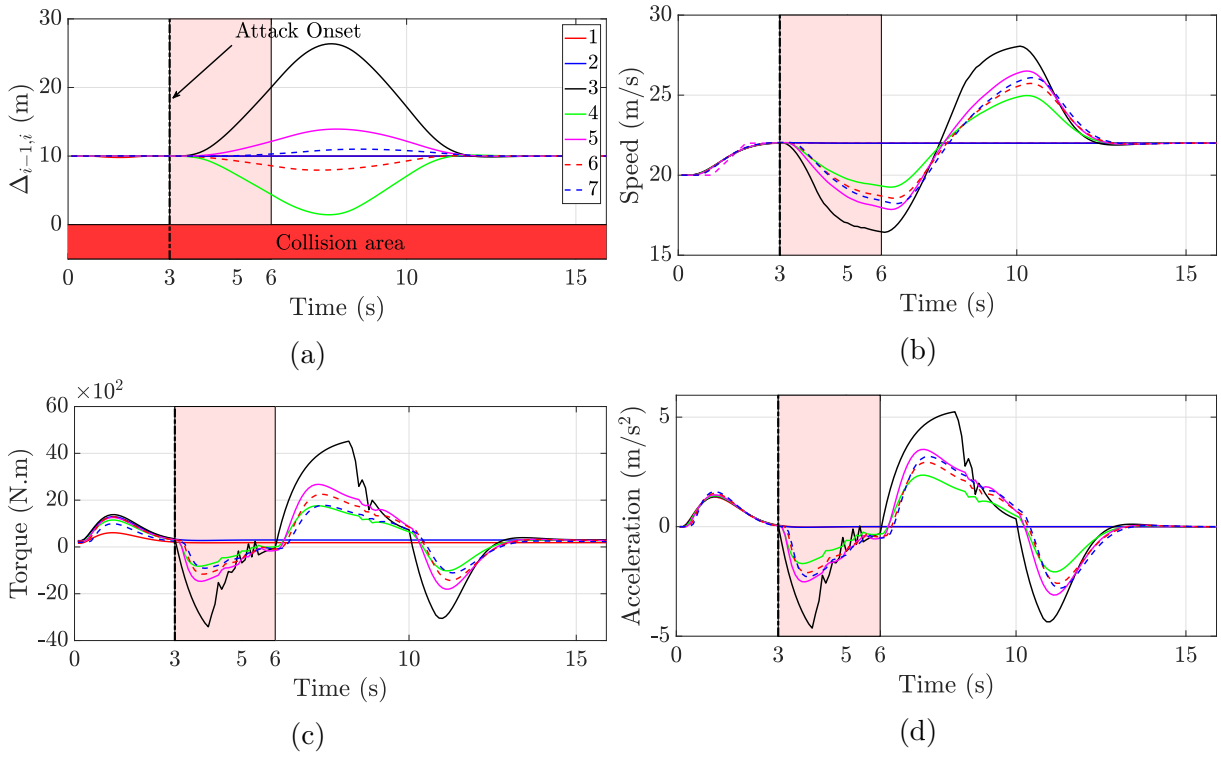


Figure 5.4: (a) Consecutive spacing, (b) speed, (c) torque, and (d) acceleration of the TPF heterogeneous DoS attacked platoon

to a DoS attack. Hence, vehicle 3 will not be able to receive the real time data including the position and velocity of vehicle 1 while the attack is performing (see Fig. 5.3). It is notable that to tackle a practical scenario, based on Assumption 5.1, the external intruder is only able to cause communication degradation among the vehicles for a finite time period. In the simulations the DoS attacker starts jamming the communication link from vehicle 1 to 3 for $\tau_a = 3$ seconds in the time interval $t \in [3, 6]$. Seven different vehicles with practical parameters form the platoon wherein the leader vehicle starts driving at $v_0(0) = 20m/s$ for one second, then it accelerates to reach $v_0(2) = 22m/s$ and continues with this velocity until the end of the simulation. The parameters of the participating vehicles in the platoon are listed in Table 5.1 which is in accordance with [184]. The prediction horizon and desired spacing among consecutive vehicles have been chosen as $N_p = 20$, and $\Delta_0 = 10$ meters, respectively. We have extended the code in [185] for our security analysis. From Fig. 5.4a one can see that despite the blockage of data transfer link from vehicle 1 to 3, there is no collision occurred in the platoon and the safety has

Table 5.1: Parameters of the participating vehicles in the static platoon

Vehicle index	m_i (kg)	τ_i (sec)	$C_{A,i}$ (N sec^2 m^{-2})	r_i (m)
1	1035.7	0.51	0.99	0.30
2	1849.1	0.75	1.15	0.38
3	1934.0	0.78	1.17	0.39
4	1678.7	0.70	1.12	0.37
5	1757.7	0.73	1.13	0.38
6	1743.1	0.72	1.13	0.37
7	1392.2	0.62	1.06	0.34

been ensured. Besides, Fig. 5.4b demonstrates that Secure–DNMPC algorithm effectively mitigates the DoS attack and the followers begin to keep tracking the leader's speed profile shortly after the attack is over. Convergence of torque and acceleration are also shown in Fig. 5.4c, and 5.4d. We highlight that the proposed algorithm has been also successfully tested on different platoon formations such as Two-Predecessor Leader Follower (TPLF), with different spacing policies such as Constant Time Headway (CTH) policy, and also on Federal Test Procedure (FTP) drive cycle to emulate urban driving.

Remark 5.8. To select an appropriate value for the prediction horizon, one has to notice as τ_a increases, N_p needs to be decreased in order to let the vehicles have enough time to exchange and update their data prior to the attack occurrence. On the other hand, too small values for N_p results in frequent rapid oscillations in the control input which makes the controller unimplementable in practice. □

5.5 Summary

Having focused on a general static heterogeneous platoon formation under the risk of DoS attack, we proposed a secure control algorithm which enables the platoon to detect and mitigate the devastation imposed by the intruder. The algorithm guarantees the desired platoon performance in terms of its control objectives together with its safety. Besides, the proposed method tackles the attacker regardless of the employed communication topology among the vehicles. Simulations performed on a TPF heterogeneous platoon with practical vehicle dynamics parameters, indicate the efficacy of the proposed technique. Dealing with other cyber attacks along with achieving additional performance objectives such as driving comfort and ecological driving are left as our future studies.

Chapter 6

Attack Resilient Nonlinear Heterogeneous Vehicle Platooning: Dynamic Case

Vehicles participating in a realistic platoon most likely bear variant nonlinear dynamics forming a heterogeneous platoon. It has been widely proved in the literature that nonlinear control techniques are mandatory to achieve desired formation objectives, such as maintaining a safe gap among consecutive cars while tracking the speed profile of the leader vehicle. As was mentioned before, despite the benefits arisen from wireless connectivity among these vehicles, this makes the whole system susceptible to cyber attacks. Recalling from the previous chapter, one such a prevalent attack, that has broadly drawn the attention of both cyber-security and control communities, is DoS attack. A DoS intelligent intruder aims at jamming communication links among cars through overwhelming the beacon node by fake requests, hence, hinders the network from processing legitimate requests. This can result in huge performance degradation and even hazardous collisions. In the literature, lack of a systematic approach adhering to control performance objectives of a dynamic nonlinear heterogeneous platoon while mitigating the DoS attack effects is yet sensible. Thus, in this chapter, we focus on an attacked dynamic nonlinear heterogeneous platoon in which arbitrary vehicles might perform cut-in/cut-out maneuvers. Variant nonlinear dynamics of the participating cars are considered in the model to form a realistic nonlinear heterogeneous platoon. Here, we extend the given approach in the previous chapter which ensures the desired control performance of a dynamic nonlinear heterogeneous platoon equipped by different communication topologies under the premise of the existence of DoS attack. The proposed method is capable of providing safe and secure control of dynamic platoons

in which arbitrary vehicles might perform cut-in and/or cut-out maneuvers. Convergence time analysis of the system are also investigated. Furthermore, to handle DoS attacks modeled by an exceeding time delay in inter-vehicular data transmission, we propose the integration of an Unscented Kalman Filter (UKF) design within the controller resulting in a novel Secure–DNMPC–UKF co-design. This, in essence, estimates the delayed system states and feeds the predicted values to the Secure–DNMPC, which efficiently mitigates the attack effects. Simulation results demonstrate the fruitfulness of the proposed method.

Contributions of this chapter are explicitly as follows. Under the premise of existence of a DoS attacker of either a network blocker or a huge time delay injector, we propose a Secure Distributed Nonlinear Model Predictive Control (Secure–DNMPC) framework to detect and mitigate the attack effects while ensuring fulfillment of the platoon control objectives. The algorithm is flexible to adopt different communication topologies handling inter-vehicular data transfer among the vehicles. Convergence time and stability analysis of the algorithm is proved in some cases. Furthermore, in case of a DoS attacker as an exceeding time delay injector, since the transferred data are still available while the attack is underway, we propose to make use of the outdated system states and take benefit of them to implement the control strategy instead of simply ignoring the data and using the most recent one. This will effectively improve the control performance of the whole system. In essence, we propose to embed a UKF as the state observer within the design of the Secure–DNMPC to adapt the algorithm to the delayed data transmission. This results in a novel Secure–DNMPC–UKF co-design. In addition, this gives the opportunity to either consider non-ideal noisy sensors or take into account the contaminated sent data due to the noisy surrounding environment and road conditions.

The results of this chapter have been published in [186, 187].

6.1 Organization of the Chapter

The remainder of this chapter is organized as follows. Sec. 6.2 presents the system modeling, including the platoon model and different types of DoS attack descriptions. Sec. 6.3 details the design of the secure controller together with some stability analysis results. Adaptation of the algorithm to handle dynamic maneuvers together with convergence time analysis are given in Sec. 6.4. Sec. 6.5 demonstrates the simulation results on a Two-Predecessor Follower (TPF) attacked nonlinear dynamic heterogeneous platoon. Finally, Sec. 6.6 concludes the chapter.

6.2 System Modeling

6.2.1 Platoon Model

Let us consider a platoon of vehicles, consisting of a Leader Vehicle (LV) and N Follower Vehicles (FVs) indexed by $\mathcal{N} := \{1, \ldots, N\}$. In this chapter, we consider the longitudinal dynamics and unidirectional communication topologies. Let Δt be the discrete time interval and $p_i(t)$, $v_i(t)$, and $T_i(t)$ denote the position, velocity, and the integrated driving/breaking torque of the i-th FV at time t, respectively. For the i-th FV, we denote the vehicle mass, the coefficient of aerodynamic drag, the coefficient of rolling resistance, the inertial lag of longitudinal dynamics, the tire radius, the mechanical efficiency of the driveline, and the control input by m_i, $C_{A,i}$, $f_{r,i}$, τ_i, r_i, η_i, and $u_i(t) \in \mathbb{R}$, respectively and g is the gravity constant. The dynamics of the i-th FV can be stated via the following discrete-time nonlinear model [61]

$$\begin{cases} \boldsymbol{x}_i(t+1) = \boldsymbol{\phi}_i(\boldsymbol{x}_i(t)) + u_i(t)\, \boldsymbol{\psi}_i \\ \quad \boldsymbol{y}_i(t) = \boldsymbol{\gamma}\, \boldsymbol{x}_i(t), \end{cases} \tag{6.1}$$

where $\boldsymbol{x}_i(t) := [p_i(t), v_i(t), T_i(t)]^\top \in \mathbb{R}^3$ and $\boldsymbol{y}(t) := [p_i(t), v_i(t)]^\top \in \mathbb{R}^2$ are the states and outputs of each vehicle, respectively. Also, $\boldsymbol{\psi}_i := [0, 0, (1/\tau_i)\, \Delta t]^\top$, $\boldsymbol{\gamma} := \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$, and

$$\boldsymbol{\phi}_i(\boldsymbol{x}_i(t)) := \begin{bmatrix} p_i(t) + v_i(t)\, \Delta t \\ v_i(t) + \frac{\Delta t}{m_i} \left(\frac{\eta_i}{r_i} T_i(t) - C_{A,i}\, v_i^2(t) - m_i\, g\, f_{r,i} \right) \\ T_i(t) - (1/\tau_i)\, T_i(t)\, \Delta t \end{bmatrix}. \tag{6.2}$$

Stacking the states, outputs, and the control input signals of all vehicles into vectors yields the platoon dynamics as follows

$$\begin{cases} X(t+1) = \boldsymbol{\Phi}(X(t)) + \boldsymbol{\Psi} U(t), \\ Y(t+1) = \boldsymbol{\Theta} \cdot X(t+1), \end{cases} \tag{6.3}$$

where $X(t) = [x_1(t)^\mathsf{T}, x_2(t)^\mathsf{T}, \ldots, x_N(t)^\mathsf{T}]^\mathsf{T} \in \mathbb{R}^{3N \times 1}$, $Y(t) = [y_1(t)^\mathsf{T}, y_2(t)^\mathsf{T}, \ldots, y_N(t)^\mathsf{T}]^\mathsf{T} \in \mathbb{R}^{2N \times 1}$, $U(t) = [u_1(t), u_2(t), \ldots, u_N(t)]^\mathsf{T} \in \mathbb{R}^{N \times 1}$. Besides, $\boldsymbol{\Phi} = [\phi_1^\mathsf{T}, \phi_2^\mathsf{T}, \ldots, \phi_N^\mathsf{T}]^\mathsf{T} \in \mathbb{R}^{3N \times 1}$, $\boldsymbol{\Psi} = \mathrm{diag}\{\psi_1, \psi_2, \ldots, \psi_N\} \in \mathbb{R}^{3N \times N}$, and $\boldsymbol{\Theta} = I_N \otimes \gamma \in \mathbb{R}^{2N \times 3N}$.

Let $\boldsymbol{\mathcal{A}} = [a_{ij}] \in \mathbb{R}^{N \times N}$ be the adjacency matrix of the underlying platoon graph topology where $a_{ij} = 1$ $(= 0)$ means that the j-th FV can (cannot) send information to the i-th

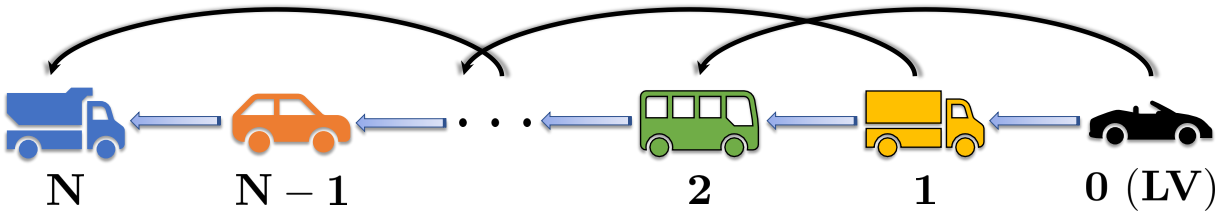


Figure 6.1: TPF heterogeneous vehicle platoon with a leader and N followers

FV, and $\boldsymbol{\mathcal{D}} = \text{diag}\{\deg_1, \deg_2, \ldots, \deg_n\}$ be the degree matrix, where $\deg_i = \Sigma_{j=1}^{n} a_{ij}$. Also, let $p_i = 1$ $(= 0)$ mean that the i-th FV is (not) pinned to the LV and gets (does not get) information from it. Suppose $\mathbb{P}_i := \{0\}$ if $p_i = 1$ and $\mathbb{P}_i := \varnothing$ if $p_i = 0$. The pinning matrix is then defined by $\boldsymbol{\mathcal{P}} = \text{diag}\{p_1, p_2, \ldots, p_n\}$. We denote $\mathbb{N}_i := \{j | a_{ij} = 1, j \in \mathcal{N}\}$ and $\mathbb{O}_i := \{j | a_{ji} = 1, j \in \mathcal{N}\}$ as the sets of FVs which the i-th FV can get information from and send information to, respectively. The set $\mathbb{I}_i := \mathbb{N}_i \cup \mathbb{P}_i$ is the set of all vehicles sending information to the i-th FV. In this study, for convenience, we consider a dynamic heterogeneous platoon equipped by Two-Predecessor Follower (TPF) communication topology shown in Fig. 6.1; however, it is straightforward to adapt our algorithm to other communication topologies.

Assumption 6.1. The directed graph of the platoon topology contains a spanning tree rooted at the LV. This assumption is necessary for stability in both homogeneous [188] and heterogeneous [61] platooning. This ensures that all vehicles get the leader's information either directly or indirectly. □

6.2.2 Platoon Control Objectives

The control objectives of the platoon are to track the speed profile generated by the leader while keeping the safe desired distance between the vehicles. Mathematically, we aim at $\lim_{t\to\infty} |v_i(t) - v_0(t)| = 0$ and $\lim_{t\to\infty} |p_{i-1}(t) - p_i(t) - d| = 0$ where d is the desired constant distance between every two consecutive vehicles. We also denote the distance between the i-th and j-th FVs by $d_{i.j}$.

Two types of output are considered here, which are the predicted and assumed outputs. The former is obtained by the calculated control input from optimization, which is fed to the system. The latter is obtained by shifting the optimal output of the last-step optimization problem. Let $\boldsymbol{y}_i^p(k|t)$ and $\boldsymbol{y}_i^a(k|t)$ denote the predicted output and the assumed output, respectively. We explain the details of how to obtain these two outputs in the

following sections. The predicted and assumed states are denoted by $\boldsymbol{x}_i^p(k|t)$ and $\boldsymbol{x}_i^a(k|t)$, respectively.

6.2.3 Attack Description

Extending the results of the previous chapter, we mainly focus on a widespread cyber-attack, called the DoS attack modeled by two different approaches used in the literature. Basically, endangering the security of the system, a DoS attacker jams the network by flooding it with fake requests such that the shared network gets overwhelmed by these demands; hence, becomes too busy to process the legitimate requests sent by the authorized users [105, 179]. This inherently causes packet loss or at least suffering delays in data transfers. In our application, we study two different DoS attack modeling introduced in the literature, i.e.,

- The DoS attacker is able to block the communication link among two nonconsecutive neighboring vehicles, which results in missing inter-vehicular data received by the follower vehicle. In essence, if the communication link among vehicle i and $i-2$ is attacked during $t \in [t_0, t_1]$, the vehicle i is only able to receive the valid data up to $t = t_0$ and has the exact same data until the attack is over, i.e., vehicle i will restart to receive updated data from vehicle $i-2$ at $t > t_1$. In the rest of the chapter, we denote $\tau_a = t_1 - t_0$ as the attack period for notational convenience.

- Another prevalent DoS attack type is to view the intruder as who injects a relatively large delay in the data transmission network. Hence, in this case, during the attack period τ_a, the follower vehicles receive the data with the time delay τ_r. This time delay is much larger compared to a threshold for a practical DSRC network.[1] The threshold can be calculated based on the acceptable probability of false alarm rate P_{FAR}

$$P_{\text{FAR}} = \int_{\delta}^{\infty} \Gamma(\tau) d\tau \leq P_{\text{FAR, acceptable}}, \tag{6.4}$$

  where $\Gamma(\tau)$ is the probability density function of the time delay, and δ is the chosen threshold (shown in Fig. 6.2) [129]. The threshold δ can also be determined using Monte-Carlo simulations, False Positives and True Negatives [109].

We will propose countermeasures in subsequent sections to face both of the aforementioned attack modelings.

[1] It should be noted that the acceptable time delay heavily depends on the application. Here, we focus on the automotive control application.

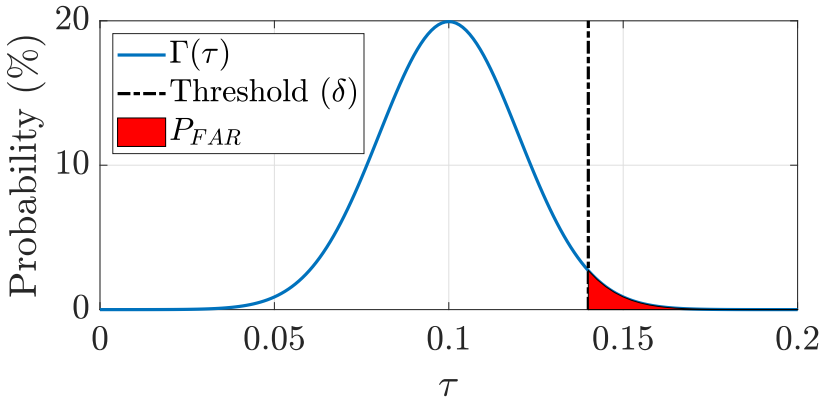


Figure 6.2: Probability distribution function of the false alarm rate and the threshold

6.3 Secure Controller Design for Dynamic Heterogeneous Platooning

6.3.1 Overview

To countermeasure the DoS attacker explained in the previous section, we take advantage of a modified version of the Distributed Nonlinear Model Predictive Control (DNMPC) approach proposed in [61], called Secure–DNMPC which aims at mitigating the effects of the attack while achieving the desired control objectives. The algorithm basically consists of two main phases, namely i) detection and ii) mitigation phase. In the first phase, we seek to detect if a DoS attack is running. If an attack is detected and the attacked communication link corresponds to the ego-vehicle with its immediate preceding or following vehicle, then the algorithm ignores the data received through the V2V link (until the attack is over) and switches to the on-board sensors followed by the implementation of the DNMPC. Otherwise, if the blocked link corresponds to the farther neighbors of the ego-vehicle, the victim vehicle makes use of the most recent updated data prior to the attack commence, and the mitigation phase starts by performing Secure–DNMPC. Inherently, in the second phase, each vehicle solves a local optimal control problem detailed as follows to generate its own optimal control input signal, which is used to compute the assumed states. The assumed states are then exchanged with the neighbors. Moreover, if the intruder targets the communication link by injecting a huge amount of time delay in data transmission, denoted by τ_r, the algorithm switches to the Secure–DNMPC–UKF mode to make use of the delayed states as much as possible. Specifically, in this case, the controller employs the observer to estimate the delayed states and provides the controller with the

predicted data. The mechanism of the controller in both of the above-mentioned cases are detailed in the following sections.

Assumption 6.2. As a standard assumption and from a practical point of view, we assume that the attacker has a limited resource of energy preventing him from jamming the network ceaselessly [111, 180, 181]. □

Remark 6.3. It is notable that the DoS attacker never attacks a link between two consecutive vehicles. The reason is that in the algorithm, the positions and velocities are transmitted, which can be reliably measured by on-board sensors mounted on an ego-vehicle such as GPS and radar. Thus, once a follower detects that those quantities are no longer updated, it can switch to its redundant sensors to obtain real time data. □

6.3.2 Design of the Secure Controller

Similar to the previous chapter, let us consider a predictive horizon N_p for the model predictive control employed to control the platoon. Suppose the predicted control inputs over the horizon are $\mathcal{U}_i^p(t-\tau_a) := \{u_i^p(0|t-\tau_a), \dots, u_i^p(N_p-1|t-\tau_a)\}$ which need to be calculated by the following optimization problem, which is the local NMPC problem that each vehicle needs to solve at each time instant t

$$
\begin{aligned}
\underset{\mathcal{U}_i^p(t-\tau_a)}{\text{minimize}} \quad & J_i(\boldsymbol{y}_i^p, u_i^p, \boldsymbol{y}_i^a, \boldsymbol{y}_{-i}^a) && \text{(6.5a)}\\
\text{subject to} \quad & \boldsymbol{x}_i^p(k+1|t-\tau_a) = \boldsymbol{\phi}_i(\boldsymbol{x}_i^p(k|t-\tau_a)) + u_i^p(k|t-\tau_a)\,\boldsymbol{\psi}_i, && \text{(6.5b)}\\
& \boldsymbol{y}_i^p(k|t-\tau_a) = \boldsymbol{\gamma}\,\boldsymbol{x}_i^p(k|t-\tau_a), && \text{(6.5c)}\\
& \boldsymbol{x}_i^p(0|t-\tau_a) = \boldsymbol{x}_i(t-\tau_a), && \text{(6.5d)}\\
& u_i^p(k|t-\tau_a) \in \mathfrak{U}_i, && \text{(6.5e)}\\
& \boldsymbol{y}_i^p(N_p|t-\tau_a) = \frac{1}{|\mathbb{I}_i|}\sum_{j\in\mathbb{I}_i}\big(\boldsymbol{y}_{-j}^a(N_p|t-\tau_a) + \widetilde{\boldsymbol{d}}_{i,j}\big), && \text{(6.5f)}\\
& T_i^p(N_p|t-\tau_a) = h_i(v_i^p(N_p|t-\tau_a)), && \text{(6.5g)}
\end{aligned}
$$

where $\boldsymbol{y}_{-i}(t) := [\boldsymbol{y}_{i_1}^\top, \dots, \boldsymbol{y}_{i_m}^\top]^\top$ (if $\{i_1, \dots, i_m\} := \mathbb{N}_i$), $\mathfrak{U}_i = \{u_i \mid u_i \in [\underline{u}_i, \bar{u}_i]\}$ defines the feasible bounds on the control input, $|\mathbb{I}_i|$ is the cardinality of $\mathbb{I}_i$, $\widetilde{\boldsymbol{d}}_{i,j} := [d_{i.j}, 0]^\top$, and τ_a is the duration of the attack. The last two terminal constraints are to make the DNMPC algorithm stable. Detailed description of the above constraints are the same as the previous chapter and is not repeated here for the sake of brevity.

The objective function (6.5a) is defined as the summation of all local cost functions

$$
\begin{aligned}
J_i(\boldsymbol{y}_i^p, u_i^p, \boldsymbol{y}_i^a, \boldsymbol{y}_{-i}^a) := \sum_{k=0}^{N_p-1} \Big(& \|\boldsymbol{y}_i^p(k|t-\tau_a) - \boldsymbol{y}_{\mathrm{des},i}(k|t-\tau_a)\|_{\boldsymbol{Q}_i} \\
& + \|u_i^p(k|t-\tau_a) - h_i(v_i^p)\|_{R_i} + \|\boldsymbol{y}_i^p(k|t-\tau_a) - \boldsymbol{y}_i^a(k|t-\tau_a)\|_{\boldsymbol{F}_i} \\
& + \sum_{j\in\mathbb{N}_i} \|\boldsymbol{y}_i^p(k|t-\tau_a) - \boldsymbol{y}_{-j}^a(k|t-\tau_a) - \widetilde{\boldsymbol{d}}_{i,j}\|_{\boldsymbol{G}_i} \Big),
\end{aligned}
\tag{6.6}
$$

in which, for a weight matrix $\boldsymbol{A} \succeq 0$, $\|\boldsymbol{x}\|_{\boldsymbol{A}} := \boldsymbol{x}^\top \boldsymbol{A}\, \boldsymbol{x}$. In (6.6), $0 \preceq \boldsymbol{Q}_i, \boldsymbol{F}_i, \boldsymbol{G}_i \in \mathbb{R}^2$ and $0 \leq R_i \in \mathbb{R}$ are the weight matrices which are the NMPC regularization factors. In fact, the matrices $\boldsymbol{Q}_i$, R_i, $\boldsymbol{F}_i$, $\boldsymbol{G}_i$ penalize for deviation of the predicted output from the desired output $\boldsymbol{y}_{\mathrm{des},i}(k|t-\tau_a)$, deviation of the predicted control input from the equilibrium, deviation of the predicted output from the assumed output, and deviation of the predicted output from the neighbors' assumed trajectories, respectively. For the i-th FV, the desired state and control signal are $\boldsymbol{x}_{\mathrm{des},i}(t) := [p_{\mathrm{des},i}(t), v_{\mathrm{des},i}(t), T_{\mathrm{des},i}(t)]^\top$ and $u_{\mathrm{des},i}(t) := T_{\mathrm{des},i}(t)$, respectively, where $p_{\mathrm{des},i}(t) := p_0(t) - i\,d$, $v_{\mathrm{des},i}(t) := v_0$, $T_{\mathrm{des},i}(t) := h_i(v_0)$ where $h_i(v_0) := (r_i/\eta_i)(C_{A,i}\, v_0^2 + m_i\, g\, f_{r,i})$ is the external drag. The desired output is $\boldsymbol{y}_{\mathrm{des},i}(t) := \boldsymbol{\gamma}\, \boldsymbol{x}_{\mathrm{des},i}(t) \in \mathbb{R}^2$.

Having injected the DoS attack on the communication network of two nonconsecutive vehicles, the follower car fails to receive updated data from its neighbor. It should be noted that what distinguishes the intelligent intruder from an intrinsic network time delay is that the data received after a huge time delay is no longer useful to generate the correct control input. To combat this attacker, we propose to integrate an Unscented Kalman Filter (UKF) within our Secure–DNMPC such that the receiver can estimate the missing data and feed the predicted values to the NMPC controller. Consequently, the NMPC controller ignores the delayed states and makes use of the predicted values as long as the attack is running. We refer to this mode of the controller as the Secure–DNMPC–UKF mode. The controller is then switched back to Secure–DNMPC once either the attack is over or the injected time delay falls below the specified threshold.

Embedding the UKF within our design takes us one more step closer to a more realistic vehicle platoon system. In particular, through our proposed co-design, we can take the process and sensor noise into account as well, which is of high importance especially for measurement sensors. From one side, assuming ideal non-noisy sensors, as done in most of the existing works in the literature, is a contrived assumption. On the other hand, the signals sent through the environment from one vehicle to another will be most likely compromised by some noise due to the surrounding environment and road conditions.

The reason of choosing UKF over Extended Kalman Filter (EKF) is to avoid the propagation of the state distribution approximation error through the system dynamics caused by the first-order linearization performed in EKF. This is vital in terms of ensuring the safety of the platoon as the propagated error in the true posterior mean and covariance of the transformed Gaussian random variable may be large and cause unsafe driving behavior. Remarkably, adopting UKF does not impose anymore computational burden compared to EKF. The interested reader is referred to [189] for more details on the superiority of UKF over EKF for nonlinear state estimation. Fig. 6.3 shows a flowchart illustrating the procedure of the Secure–DNMPC–UKF co-design.

Before delving into the details of the Secure–DNMPC–UKF algorithm, a quick overview of the basics of Unscented Kalman Filtering are given in the following.

6.3.3 Principles of Unscented Kalman Filtering

Unscented Kalman Filter, as a nonlinear state observer, basically relies on the unscented transformation to capture the statistical properties of state estimates via nonlinear functions. The observer initially captures the mean and covariance of the state estimates through a set of so-called sigma points. The algorithm makes use of those sigma points as the inputs of the process and measurement functions to generate a new set of states. Subsequently, a set of state estimates and state estimation error covariance are obtained using the mean and covariance of the previously transformed points.

Let us consider an n-state nonlinear system described by the following nonlinear state transition and measurement functions compromised by additive zero-mean process noise $\boldsymbol{w}[k] \sim (0, \boldsymbol{Q}[k])$ and measurement noise $\boldsymbol{v}[k] \sim (0, \boldsymbol{R}[k])$

$$\begin{cases} \boldsymbol{x}[k+1] = f(\boldsymbol{x}[k], u_s[k]) + \boldsymbol{w}[k] \\ \quad \boldsymbol{y}[k] = h(\boldsymbol{x}[k], u_m[k]) + \boldsymbol{v}[k] \end{cases} \tag{6.7}$$

The filter takes the following steps to obtain the state estimates and the state estimation error covariance

1. The filter is initialized with an initial value for state $\boldsymbol{x}[0]$ and state estimation error covariance matrix P

$$\hat{\boldsymbol{x}}[0|-1] = \mathbb{E}(\boldsymbol{x}[0]) \tag{6.8}$$

$$P[0|-1] = \mathbb{E}[(\boldsymbol{x}[0] - \hat{\boldsymbol{x}}[0]-1)(\boldsymbol{x}[0] - \hat{\boldsymbol{x}}[0]-1)^\top] \tag{6.9}$$

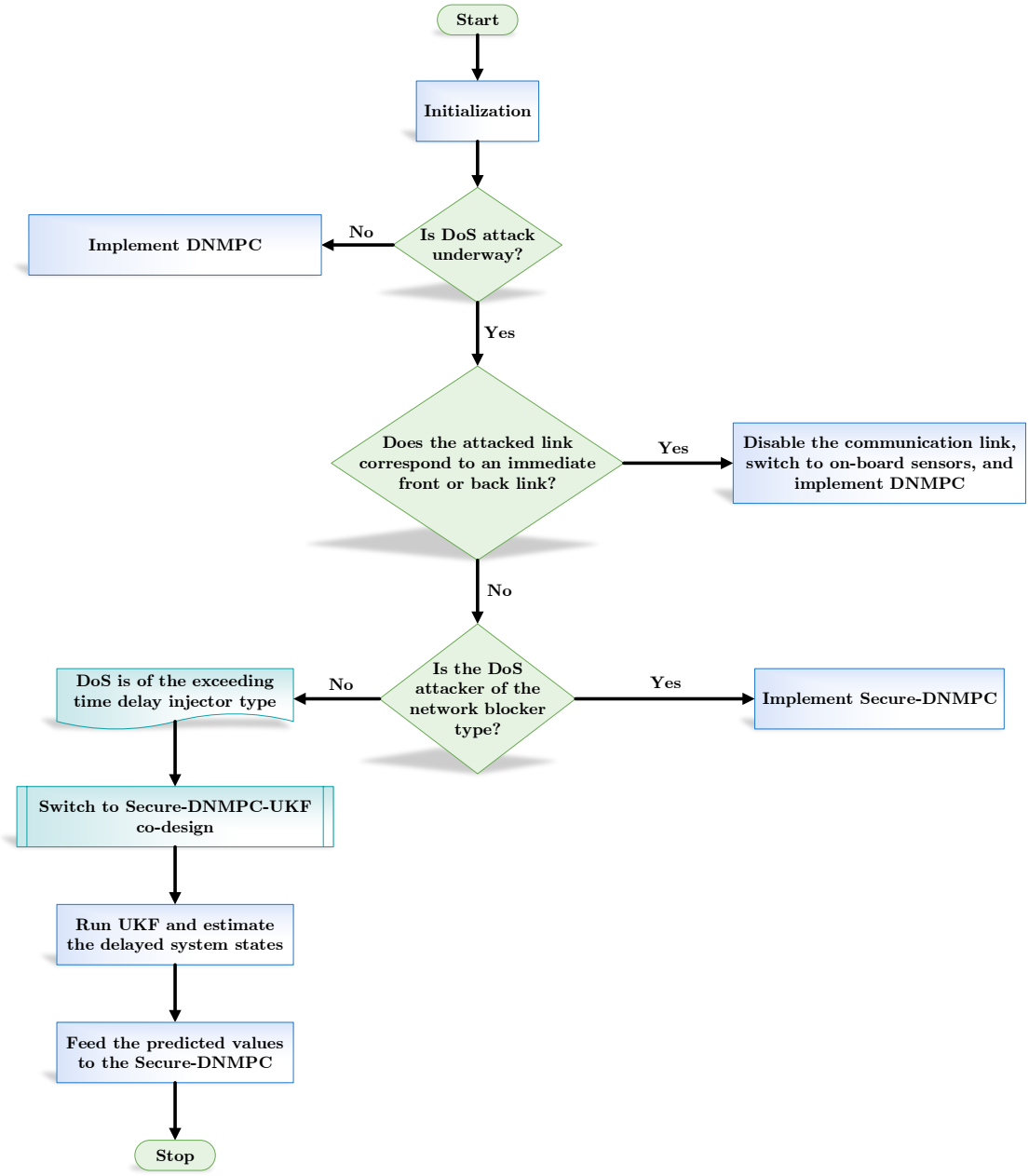


Figure 6.3: Procedure of the proposed Secure–DNMPC–UKF co-design

where $\hat{\boldsymbol{x}}[k]$ is the state estimate at time k and $\hat{\boldsymbol{x}}[k_1|k_0]$ denotes the state estimate at time k_1 using the measurement data up to time k_0.

2. Having used the measurement data $\boldsymbol{y}[k]$ at each time instant k, the filter updates the state estimate and the state estimation error covariance:

   (a) Choose the sigma points $\hat{\boldsymbol{x}}^{(i)}[k|k-1]$ at time k

$$\hat{\boldsymbol{x}}^{(0)} = \hat{\boldsymbol{x}}[k|k-1] \tag{6.10}$$

$$\hat{\boldsymbol{x}}^{(i)}[k|k-1] = \hat{\boldsymbol{x}}[k|k-1] + \Delta\boldsymbol{x}^{(i)}, \quad i = 1, 2, \ldots, 2n \tag{6.11}$$

$$\Delta\boldsymbol{x}^{(i)} = (\sqrt{cP[k|k-1]})_i, \quad i = 1, 2, \ldots, n \tag{6.12}$$

$$\Delta\boldsymbol{x}^{(n+i)} = -(\sqrt{cP[k|k-1]})_i, \quad i = 1, 2, \ldots, n \tag{6.13}$$

   where $c = \alpha^2(n + \kappa)$ is a scaling factor and $(\sqrt{cP})_i$ is the i-th column of the $\sqrt{cP}$ matrix [190].

   (b) For each of the sigma points, use the nonlinear measurement function to compute the predicted measurements

$$\hat{\boldsymbol{y}}^{(i)}[k|k-1] = h(\hat{\boldsymbol{x}}^{(i)}[k|k-1], u_m[k]), \quad i = 1, 2, \ldots, 2n \tag{6.14}$$

   (c) In order to obtain the predicted measurement at time k, integrate the predicted measurements

$$\hat{\boldsymbol{y}}[k] = \Sigma_{i=0}^{2n} W_n^{(i)} \hat{\boldsymbol{y}}^{(i)}[k|k-1] \tag{6.15}$$

$$W_n^{(0)} = 1 - \frac{n}{\alpha^2(n + \kappa)} \tag{6.16}$$

$$W_n^{(i)} = \frac{1}{2\alpha^2(n + \kappa)}, \quad i = 1, 2, \ldots, 2n \tag{6.17}$$

   (d) By adding the measurement noise $\boldsymbol{R}[k]$, estimate the covariance matrix of the predicted measurement

$$P_{\boldsymbol{y}} = \Sigma_{i=0}^{2n} W_c^{(i)} (\hat{\boldsymbol{y}}^{(i)}[k|k-1] - \hat{\boldsymbol{y}}[k])(\hat{\boldsymbol{y}}^{(i)}[k|k-1] - \hat{\boldsymbol{y}}[k])^\top + \boldsymbol{R}[k] \tag{6.18}$$

$$W_c^{(0)} = (2 - \alpha^2 + \beta) - \frac{n}{\alpha^2(n + \kappa)} \tag{6.19}$$

$$W_c^{(i)} = \frac{1}{2\alpha^2(n + \kappa)}, \quad i = 1, 2, \ldots, 2n \tag{6.20}$$

   For the details on effects of parameters α, β, and κ the reader is referred to [190].

(e) Estimate the cross-covariance between $\hat{\boldsymbol{x}}[k|k-1]$ and $\hat{\boldsymbol{y}}[k]$

$$P_{\boldsymbol{xy}} = \frac{1}{2\alpha^2(n+\kappa)} \Sigma_{i=1}^{2n} (\hat{\boldsymbol{x}}^{(i)}[k|k-1] - \hat{\boldsymbol{x}}[k|k-1])(\hat{\boldsymbol{y}}^{(i)}[k|k-1] - \hat{\boldsymbol{y}}[k|k-1])^\top \tag{6.21}$$

Note that $\hat{\boldsymbol{x}}^{(0)}[k|k-1] - \hat{\boldsymbol{x}}[k|k-1] = 0$.

(f) Compute the estimated state and state estimation error covariance at time step k

$$K = P_{\boldsymbol{xy}} P_{\boldsymbol{y}}^{-1} \tag{6.22}$$

$$\hat{\boldsymbol{x}}[k|k] = \hat{\boldsymbol{x}}[k|k-1] + K(\boldsymbol{y}[k] - \hat{\boldsymbol{y}}[k]) \tag{6.23}$$

$$P[k|k] = P[k|k-1] - K P_{\boldsymbol{y}} K_k^\top \tag{6.24}$$

where K is the Kalman gain.

3. Now the state and state estimation error covariance can be predicted at time instant $k+1$

(a) Choose the sigma points $\hat{\boldsymbol{x}}^{(i)}[k|k]$ at time instant k.

$$\hat{\boldsymbol{x}}^{(0)}[k|k] = \hat{\boldsymbol{x}}[k|k] \tag{6.25}$$

$$\hat{\boldsymbol{x}}^{(i)}[k|k] = \hat{\boldsymbol{x}}[k|k] + \Delta\boldsymbol{x}^{(i)}, \quad i = 1, 2, \ldots, 2n \tag{6.26}$$

$$\Delta\boldsymbol{x}^{(i)} = (\sqrt{cP[k|k]})_i, \quad i = 1, 2, \ldots, n \tag{6.27}$$

$$\Delta\boldsymbol{x}^{(n+i)} = -(\sqrt{cP[k|k]})_i, \quad i = 1, 2, \ldots, n \tag{6.28}$$

$$\tag{6.29}$$

(b) In order to get the predicted states at time $k+1$, combine the predicted states

$$\hat{\boldsymbol{x}}[k+1|k] = \Sigma_{i=0}^{2n} W_n^{(i)} \hat{\boldsymbol{x}}^{(i)}[k+1|k] \tag{6.30}$$

$$W_n^{(0)} = 1 - \frac{n}{\alpha^2(n+\kappa)} \tag{6.31}$$

$$W_n^{(i)} = \frac{1}{2\alpha^2(n+\kappa)}, \quad i = 1, 2, \ldots, 2n \tag{6.32}$$

4. To account for the process noise, add $\boldsymbol{Q}[k]$ and compute the covariance of the pre-

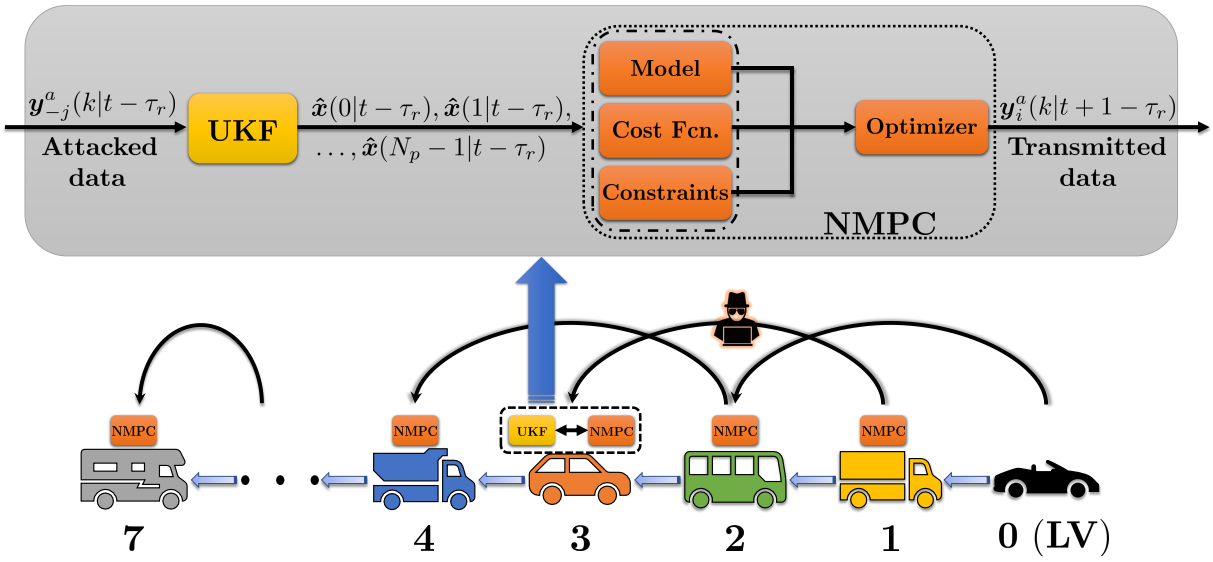


Figure 6.4: Schematic of the proposed Secure–DNMPC–UKF co-design

dicted state

$$P[k+1|k] = \Sigma_{i=0}^{2n} W_c^{(i)} (\hat{\boldsymbol{x}}^{(i)}[k+1|k] - \hat{\boldsymbol{x}}[k+1|k])(\hat{\boldsymbol{x}}^{(i)}[k+1|k] - \hat{\boldsymbol{x}}[k+1|k])^\top + \boldsymbol{Q}[k] \tag{6.33}$$

$$W_c^{(0)} = (2 - \alpha^2 + \beta) - \frac{n}{\alpha^2(n+\kappa)} \tag{6.34}$$

$$W_c^{(i)} = \frac{1}{2\alpha^2(n+\kappa)}, \quad i = 1, 2, \ldots, 2n \tag{6.35}$$

For more details on the observer for the case of non-additive process/measurement noise, please see [190].

The proposed algorithm is summarized in Algorithm 3, which is the extended version of the given approach in the previous chapter for static platoons. We further note that $\boldsymbol{y}_i^a(t)$ represents the data sent by the vehicle i to the set $\mathbb{O}_i$ while $\boldsymbol{y}_{-j}^a$ denotes the data received by the vehicle i from its neighbors $j \in \mathbb{N}_i$. Superscript a, p, and $*$ are to distinguish between assumed, predicted, and optimal quantities, respectively. The assumed quantities are the ones transmitted by the vehicles in the platoon. Fig. 6.4 illustrates the Secure–DNMPC–UKF co-design in which $\hat{\boldsymbol{x}}(k|t)$ denotes the estimated state at time instant k using the measured data up to time t.

Algorithm 3 Secure–DNMPC–UKF for Dynamic Nonlinear Heterogeneous Vehicle Platooning under DoS Attack

Initialization:
Assumed values for vehicle i are set at time $t = 0$,
$u_i^a(k|0) = h_i(v_i(0)), \boldsymbol{y}_i^a(k|0) = \boldsymbol{y}_i^p(k|0), \quad k = 0, 1, \ldots, N_p - 1$
while $t \leq t_{\text{final}}$ **do**
  Cut-in/cut-out CHECK ▷ Check to see if cut-in/cut-out occurred
  Adjust data send-to/receive-from vehicles based on the occurred cut-in/cut-out
  if $p_{-j}^a(t) = p_{-j}^a(t-1), j \in \mathbb{N}_i$ **then** ▷ Check to see if a DoS is underway
    if $j = i - 1$ or $j = i + 1$ **then** ▷ Check to see if the attacked link corresponds to a predecessor or a follower
      Disable communication link, switch to on-board sensors, $\tau_a \leftarrow 0$, and **Go to:** 12
    else
      if Attacker blocks the communication link **then** ▷ Check to see if the attacker is of the blockage type
        for Each vehicle i **do** ▷ Implement Secure–DNMPC
          Solve Problem 6.5 at time $t > 0$ and yield $u_i^*(k|t-\tau_a),\ k = 0, 1, \ldots, N_p - 1$
          Compute: $\begin{cases} \boldsymbol{x}_i^*(k+1|t-\tau_a) = \boldsymbol{\phi}_i(\boldsymbol{x}_i^*(k|t-\tau_a)) + \boldsymbol{\psi}_i u_i^*(k|t-\tau_a), \\ \boldsymbol{x}_i^*(0|t-\tau_a) = \boldsymbol{x}_i(t-\tau_a), \quad k = 0, 1, \ldots, N_p - 1 \end{cases}$
          Compute: $u_i^a(k|t-\tau_a+1) = \begin{cases} u_i^*(k+1|t-\tau_a), & k = 0, 1, \ldots, N_p - 2 \\ h_i\left(v_i^*(N_p|t-\tau_a)\right), & k = N_p - 1 \end{cases}$
          Compute: $\begin{cases} \boldsymbol{x}_i^a(k+1|t-\tau_a+1) = \boldsymbol{\phi}_i\left(\boldsymbol{x}_i^a(k|t-\tau_a+1)\right) + \boldsymbol{\psi}_i u_i^a(k|t-\tau_a+1) \\ \boldsymbol{x}_i^a(0|t-\tau_a+1) = \boldsymbol{x}_i^*(1|t-\tau_a), \quad k = 0, 1, \ldots, N_p - 1 \end{cases}$
          Compute: $\boldsymbol{y}_i^a(k|t-\tau_a+1) = \boldsymbol{\gamma}\boldsymbol{x}_i^a(k|t-\tau_a+1), \quad k = 0, 1, \ldots, N_p - 1$
          Send $\boldsymbol{y}_i^a(k|t-\tau_a+1)$ to the vehicles lie in the set $\mathbb{O}_i$, and receive $\boldsymbol{y}_{-j}^a(k|t-\tau_a+1)$ from neighboring vehicles $j \in \mathbb{N}_i$ and compute $\boldsymbol{y}_{\text{des},i}(k|t-\tau_a+1)$
          Exert the first element of the optimal control signal $u_i(t-\tau_a) = u_i^*(0|t-\tau_a)$
        end for
      else if Attacker injects exceeding delay $\tau_r > \delta$ **then** ▷ Check to see if the attacker is of the exceeding time delay injector type
        Switch to Secure–DNMPC–UKF mode
        Estimate the delayed states via UKF
        Implement the Secure–DNMPC using the predicted states coming from the UKF
      end if
    end if
  end if
end while

Since the stability analysis of the controller is mostly the same as the previous chapter, we omit that here.

6.4 Dynamic Platoon Control: Handling Cut-in/Cut-out Maneuvers

In this section, we consider a dynamic heterogeneous platoon wherein arbitrary vehicle(s) might perform cut-in/cut-out maneuvers. Here, we demonstrate the ability of the proposed algorithm to handle dynamic maneuvers while the platoon is subject to the cyber attack.

First, we consider a secure dynamic heterogeneous platoon and prove some results based on which we extend the results to an insecure platoon. Assume there exist N_{ci} cut-in and N_{co} cut-out maneuvers in total while the number of initial FVs in the platoon is N. Let $\mathcal{N}_{ci} := \{1, \dots, N_{ci}\}$ and $\mathcal{N}_{co} := \{1, \dots, N_{co}\}$. We denote the time of the i-th cut-in and the j-th cut-out maneuvers by $t_{ci,i}$ and $t_{co,j}$, respectively. The following theorem determines the time of convergence of a dynamic platoon including possible cut-in and cut-out maneuvers.

Lemma 6.4 ([61, Theorem 2])**.** *If Assumption 6.1 is satisfied, then Problem (6.5) guarantees convergence of the output to the desired output in at most N time steps, i.e., $\boldsymbol{y}_i^p(N_p|t) = \boldsymbol{y}_{des,i}(N_p|t), \forall t \geq N$, for a static platoon (without any dynamic maneuvers).* □

Theorem 6.5. *When having cut-in and/or cut-out maneuvers in a secured dynamic platoon, if Assumption 6.1 is satisfied, the Problem (6.5) guarantees convergence of the output to the desired output in at most*

$$t_{conv,\ secure} := \max_{i,j}\big[t_{ci,i}, t_{co,j} \,|\, \forall i \in \mathcal{N}_{ci}, \forall j \in \mathcal{N}_{cj}\big] + N + N_{ci} - N_{co}, \tag{6.36}$$

time steps, i.e., $\boldsymbol{y}_i^p(N_p|t) = \boldsymbol{y}_{des,i}(N_p|t), \forall t \geq t_{conv,\ secure}$. □

Proof: Let $\boldsymbol{\mathcal{L}} := \boldsymbol{\mathcal{D}} - \boldsymbol{\mathcal{A}}$ be the Laplacian matrix of the underlying platoon graph topology. When a new cut-in or cut-out occurs, some new chaos is introduced to the system so we can consider the latest cut-in/cut-out maneuver. Considering the latest cut-in, one vehicle is added to the number of existing vehicles, let it be N. If the platoon graph is unidirectional and satisfies Assumption 6.1, the new $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{(N+1)\times(N+1)}$ is a lower-triangular matrix. Moreover, according to [61, Lemma 4], we have $\boldsymbol{\mathcal{D}} + \boldsymbol{\mathcal{P}} > 0$, yielding the eigenvalues of $(\boldsymbol{\mathcal{D}} + \boldsymbol{\mathcal{P}})^{-1}\boldsymbol{\mathcal{A}}$ to be zero and this matrix to be nilpotent with degree at most $N+1$. Based on [61, Lemma 1] and [61, Theorem 1], $\boldsymbol{y}_i^p(N_p|t)$ converges to the desired output in at most $N+1$ steps. Extending this to N_{ci} cut-in maneuvers requires $N + N_{ci}$ time steps after the latest cut-in. Similar analysis can be performed for the cut-out maneuvers, resulting in $N - N_{co}$ time steps after the latest cut-out because the number of vehicles has been reduced. In general, having N_{ci} cut-in and N_{co} cut-out maneuvers

will need $N + N_{ci} - N_{co}$ time steps after the latest maneuver which can be formulated as $\max_{i,j}[t_{ci,i}, t_{co,j} \,|\, \forall i \in \mathcal{N}_{ci}, \forall j \in \mathcal{N}_{cj}]$. ■

Corollary 6.6. *Lemma 6.4, for the static platoon, is a special case of Theorem 6.5 which is for a dynamic platoon.* □

Proof: When neither cut-in nor cut-out happen, the time of convergence is $t_{\text{conv, secure}} = 0 + N + 0 + 0 = N$ according to Theorem 6.5. ■

Remark 6.7 (Special Cases). Four special cases of the dynamic platoon are as follows:

1. One cut-in happens at $t = 0$ and one cut-out happens at $t = N$: According to Theorem 6.5, the platoon converges in $t = N + N + 1 - 1 = 2N$. It is correct because before the cut-out, the platoon contains $N + 1$ vehicles until time N. When cut-out happens, the platoon is changed to a platoon with N vehicles which converges in N time steps according to Lemma 6.4.

2. One cut-out happens at $t = 0$ and one cut-in happens at $t = N$: According to Theorem 6.5, the platoon converges in $t = N + N + 1 - 1 = 2N$, which is correct because in $t \in [0, N]$, the platoon includes $N - 1$ vehicles until time N. When cut-in happens, the platoon is modified to a platoon with N vehicles which converges in N time steps according to Lemma 6.4.

3. Both cut-in and cut-out happen at $t = 0$: According to Theorem 6.5, the platoon converges in $t = 0 + N + 1 - 1 = N$, which is correct because the platoon includes N vehicles which converges in N time steps according to Lemma 6.4.

4. Both cut-in and cut-out happen at $t = N$: According to Theorem 6.5, the platoon converges in $t = N + N + 1 - 1 = 2N$, which is correct because in $t \in [0, N]$, the platoon includes N vehicles. After the cut-in/cut-out actions the platoon still includes N vehicles which converges in N time steps according to Lemma 6.4.

□

Corollary 6.8. *When having cut-in and/or cut-out maneuvers in an insecure dynamic platoon, if Assumption 6.1 is satisfied, the convergence time of the output to the desired output is upper bounded by* $t_{conv,\ secure} + \max\{\tau_r, \tau_a\}$, *i.e.,*[2]

$$t_{conv,\ insecure} \leq \max_{i,j}\left[t_{ci,i}, t_{co,j} \,|\, \forall i \in \mathcal{N}_{ci}, \forall j \in \mathcal{N}_{cj}\right] + N + N_{ci} - N_{co} + \max\{\tau_r, \tau_a\}, \quad (6.37)$$

time steps, i.e., $\boldsymbol{y}_i^p(N_p|t-\tau) = \boldsymbol{y}_{des,i}(N_p|t-\tau), \forall t \geq t_{conv,\ insecure}$. □

[2]Although, this upper bound might be conservative in some cases (such as in the scenario studied in subsection 6.5.2), it provides a safe margin for the convergence time of the controller.

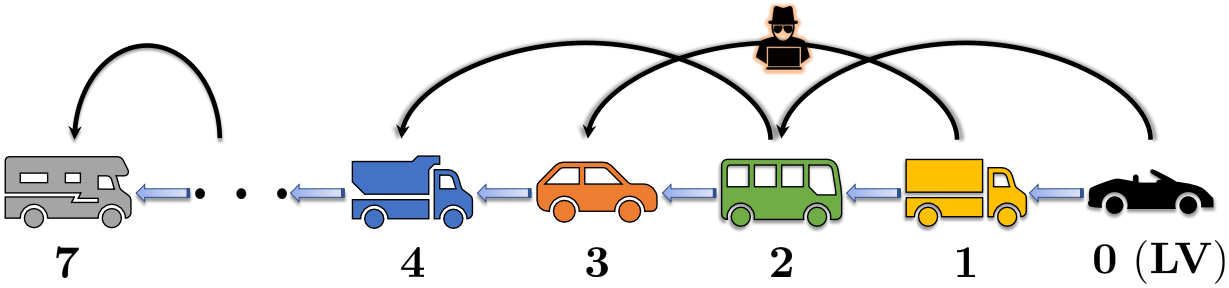


Figure 6.5: TPF heterogeneous attacked vehicle platoon with a leader and 7 followers

6.5 Simulation Results

A heterogeneous platoon consisted of seven different followers is considered where they can exchange inter-vehicular data among each other through the TPF communication topology. It is assumed that the communication link among the vehicle 1 and 3 is subject to a DoS attack. Hence, vehicle 3 will not be able to receive the real time data including the position and velocity of vehicle 1 while the attack is performing (see Fig. 6.5). It is notable that to tackle a practical scenario, based on Assumption 6.2, the external intruder is only able to cause communication degradation among the vehicles for a finite time period. In the simulations, the DoS attacker starts jamming the communication link from vehicle 1 to 3. Seven different vehicles with realistic parameters form the platoon wherein the leader vehicle starts driving at $v_0(0) = 20m/s$ for one second, then it accelerates to reach $v_0(2) = 22m/s$ and continues with this velocity until the end of the simulation. The prediction horizon and desired spacing among consecutive vehicles have been chosen as $N_p = 20$, and $d = 10$ meters, respectively. The parameters of the participating vehicles in the platoon are listed in Table 6.1 which is in accordance with [184]. We have extended the code in [185] for our security analysis.

Remark 6.9. To select an appropriate value for the prediction horizon, one has to notice as τ increases, N_p needs to be decreased in order to let the vehicles have enough time to exchange and update their data prior to the attack occurrence. On the other hand, too small values for N_p results in frequent rapid oscillations in the control input which makes the controller unimplementable in practice. □

Table 6.1: Parameters of the participating vehicles in the dynamic platoon

Vehicle index	m_i (kg)	τ_i (sec)	$C_{A,i}$ (N sec^2 m^{-2})	r_i (m)
1	1035.7	0.51	0.99	0.30
cut-in	1305.9	0.63	1.00	0.40
2	1849.1	0.75	1.15	0.38
3	1934.0	0.78	1.17	0.39
4	1678.7	0.70	1.12	0.37
5	1757.7	0.73	1.13	0.38
6	1743.1	0.72	1.13	0.37
7	1392.2	0.62	1.06	0.34

6.5.1 DoS Attack Modeled as a Network Blocker

In this part, we take one more step to effectively control the dynamic heterogeneous platoon endangered by an intelligent DoS intruder. As was previously described, the attacker could jam the communication network among any two nonconsecutive vehicle to prohibits a follower vehicle from receiving updated data. Having made an expressive scenario incorporating both cut-in and cut-out actions while taking into account a DoS attack, we consider a same setting for the attacked platoon presented in the previous section except assuming a vehicle merges with the platoon at $t = 2$sec to be placed in front of the second FV. Furthermore, we let the fourth FV to perform a cut-out action at $t = 4$sec (see Fig. 6.6). We note that the desired distance among the vehicles ($d = 10$ meters) provides enough space for a regular vehicle to cut-in. The attack happens on the communication environment among the first and third FVs in the time interval $t \in [3, 6]$. Although these tight actions might not seem to happen in practice, they are chosen to challenge the algorithm largely. Fig. 6.7 demonstrate the driving quantities of the respective platoon.

From Fig. 6.7a, one can see that despite the blockage of data transfer link from vehicle 1 to 3, there is no collision occurred in the platoon, and the safety has been ensured. Besides, Fig. 6.7b demonstrates that Secure–DNMPC algorithm effectively mitigates the DoS attack and the followers begin to keep tracking the leader's speed profile shortly after the attack is over. Convergence of torque and acceleration are also shown in Fig. 6.7c, and 6.7d. It is worth mentioning that by reducing its speed, the second FV has increased its gap with the first FV to make the desired distance of 10m for the cut-in vehicle. Consequently, the following vehicles have lessened their velocity to keep the desired distance. Fig. 6.7b verifies this fact. A similar analysis exists for the cut-out maneuver where the following

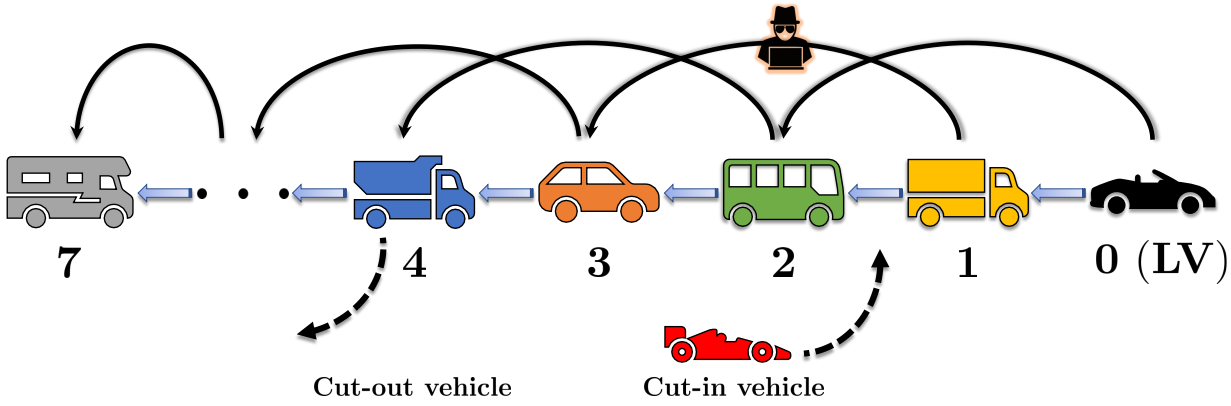


Figure 6.6: TPF dynamic heterogeneous attacked vehicle platoon with cut-in and cut-out vehicles

vehicles have increased their velocity to reach the desired distance from the vehicles in front.

As one can see, the spacing and speed tracking objectives have been safely fulfilled. To have a clearer look of the spacing objective, Fig. 6.8 shows the magnified absolute positions and the spacing error of consecutive vehicles. Since all the spacing errors in Fig. 6.8b are greater than -10 meters, no collision has occurred. Moreover, the relative spacing error shows jumps in the distance error (blue and purple curves) because of the cut-in/cut-out maneuvers.[3] As expected, the spacing error for the cut-out maneuver (purple curve in Fig. 6.8b) has an opposite sign with respect to the cut-in error (blue curve in Fig. 6.8b). Furthermore, we see that convergence has been reached in less than 14s which coincides with Corollary 6.8 because $t_{\text{conv, insecure}} \leq \max(2,4) + 7 + 1 - 1 + 3 = 14\text{s}$.

6.5.2 DoS Attack Modeled as an Exceeding Time Delay Injection in the Data Transmission

Inherent communication delay of standard 802.11p-based DSRC network ranges from tens to hundreds of milliseconds [191–193]. Here, to ensure modeling a highly devastating attacker, we assume the time delay imposed by the intruder is $\tau_r = 2.5\text{sec}$. In addition, non-ideal sensors are assumed in the simulations, i.e., an additive zero-mean white Gaussian

[3]Note that the jump in the relative spacing error of the third FV (black curve) is due to the DoS attack.

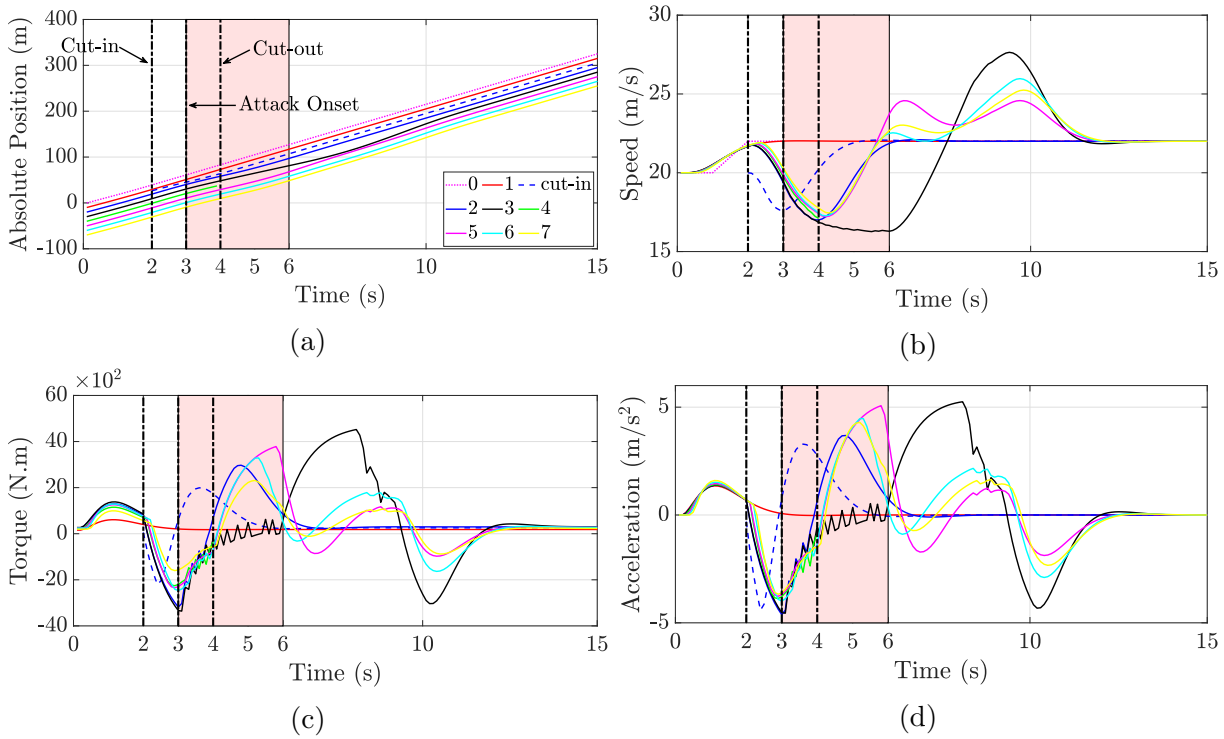


Figure 6.7: (a) Absolute position, (b) speed, (c) torque, and (d) acceleration of the TPF dynamic heterogeneous DoS attacked (network blocker) platoon with cut-in/cut-out maneuvers equipped by Secure–DNMPC

noise with variance $\sigma^2 = 0.01$ is considered on both the position and velocity sensors.[4] To challenge more the algorithm we introduce a severer attack which happens for a longer period of time, i.e., in the time range $t \in [3, 10]$. It is worth noting that cut-in and cut-out maneuvers are still in effect at $t = 2$sec and $t = 4$sec, respectively. Fig. 6.9 shows the performance of the proposed co-design controller on the attacked platoon with cut-in and cut-out actions. As is demonstrated by the driving quantities, safe distance and velocity tracking requirements have been fulfilled. Furthermore, the convergence has been reached in less than 18s which again verifies Corollary 6.8 because $t_{\text{conv, insecure}} \leq \max(2, 4) + 7 + 1 - 1 + \max(2.5, 7) = 18$s. It would also be insightful to compare the results to the case where UKF is not embedded in the design. Fig. 6.10 demonstrates the resulting driving behavior when only relying on the controller leaving out the estimation

[4]This could also be considered as the environment effects on the transmitted signals. Modeling the environment effect with white Gaussian noise in V2V communications is widely used in literature [194,195].

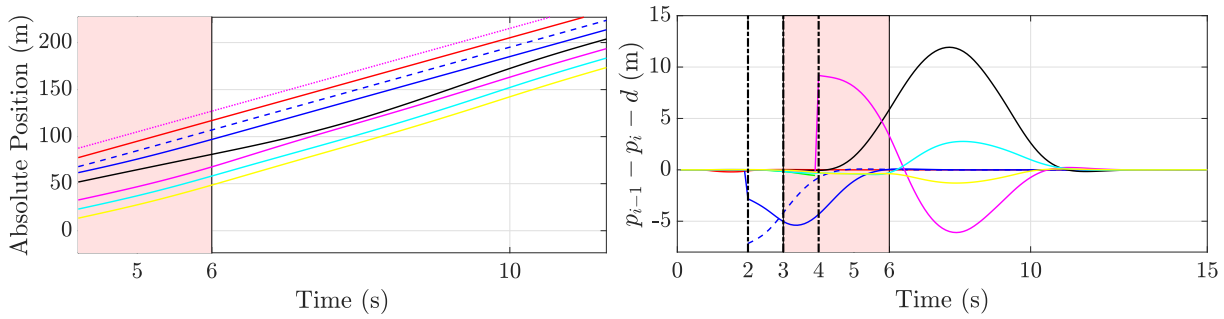


Figure 6.8: (a) Magnified absolute position and (b) spacing error of the TPF dynamic heterogeneous DoS attacked (network blocker) platoon with cut-in/cut-out maneuvers equipped by Secure–DNMPC

phase. Occurring collision and violating the control objectives clearly prove the critical role of the observer design.

It is noticeable that by comparing the previous scenarios (Figs. 6.7, 6.9, and 6.10), it reveals that embedding the UKF within our controller design, also has the advantage of reducing the oscillations in the control input caused by the cyber attack. This generation of a smoother control input enhances the driving comfort in practice.

We highlight that the proposed algorithm has also been successfully tested on different platoon formations such as Two-Predecessor Leader Follower (TPLF), with different spacing policies such as Constant Time Headway (CTH) policy, and also on Federal Test Procedure (FTP) drive cycle to emulate urban driving.

6.6 Summary

This chapter dealt with a broadly concerned control problem, namely the dynamic heterogeneous platoon control. As was explained before, a platoon is mainly consisted of networking and data transmission among the vehicles, forming the cyber layer, and the physical environment composed of the participant cars, forming the physical layer. This cyber-physical system is highly prone to cyber attacks endangering the wireless connectivity among the vehicles. This vulnerability to external attackers needs to be fully addressed as an insecure communication layer in a platoon can cause manipulated and/or missing data received by the followers resulting in dangerous hazards. In this chapter, we focused on the

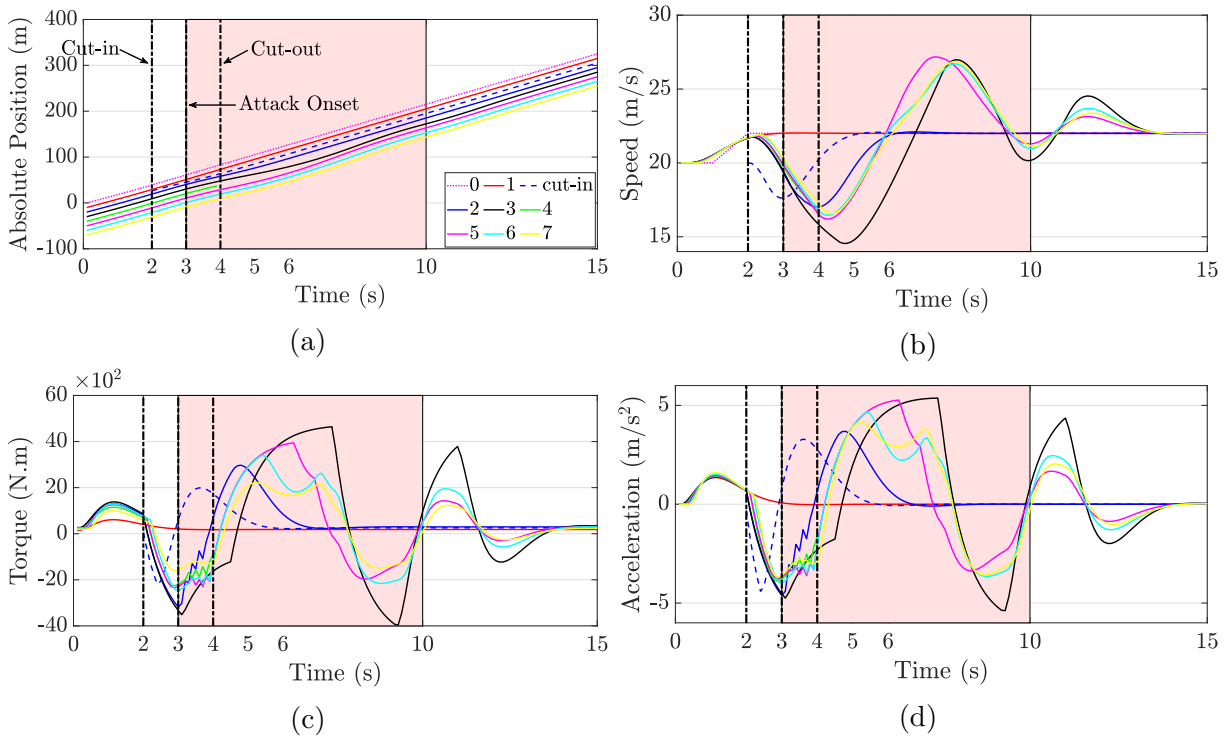


Figure 6.9: (a) Absolute position, (b) speed, (c) torque, and (d) acceleration of the TPF dynamic heterogeneous DoS attacked (exceeding time delay injector) platoon with cut-in/cut-out maneuvers equipped by Secure–DNMPC–UKF co-design

widespread so-called DoS attack in which the intelligent intruder targets the wireless links by overwhelming the node by invalid requests, hence, either blocks the network or prevents it from timely data transfer. We proposed a Secure–Distributed Nonlinear Model Predictive Control (Secure–DNMPC) framework to ensure a safe and secure dynamic platooning which fulfills both the safe distancing between the cars and speed tracking requirements. The method is capable of handling cut-in/cut-out maneuvers under the premise of the existence of a cyber DoS attack. The algorithm is basically comprised of detection and mitigation phases.

Furthermore, we introduced a novel Secure–DNMPC–UKF co-design for the case when the DoS attacker injects a huge amount of time delay in the network compared to the intrinsic practical DSRC time delay. This makes use of the available but outdated data to estimate and predict future states. The proposed approach also provides the opportunity to consider non-ideal sensors which contaminate the measured data. In addition,

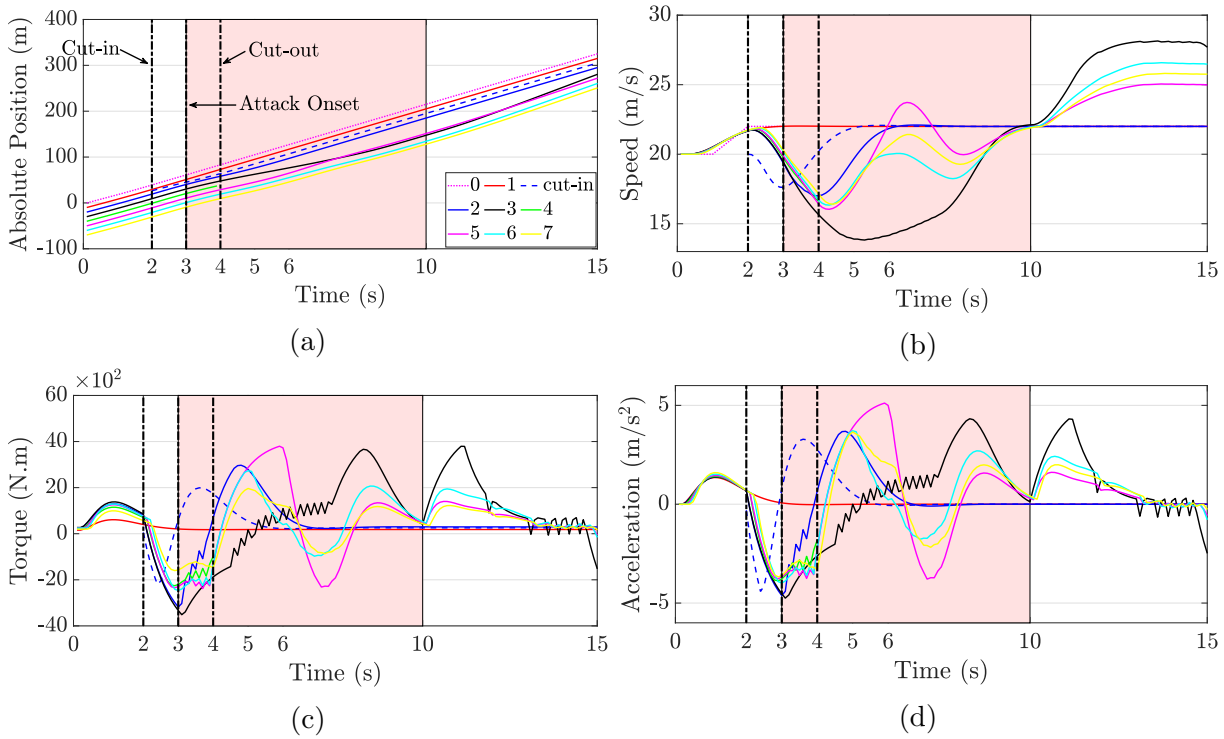


Figure 6.10: (a) Absolute position, (b) speed, (c) torque, and (d) acceleration of the TPF dynamic heterogeneous DoS attacked (exceeding time delay injector) platoon with cut-in/cut-out maneuvers without UKF design

compromised signals sent through a realistic noisy environment can be considered as well. Simulation results demonstrated the efficacy of the introduced technique. As a future direction, one can think of generalizing the given algorithm to a multi-platooning scenario in which two or more attacked platoons drive in parallel, and arbitrary vehicles wish to exit their own platoon and merge with an adjacent one. Also, other types of attacks and the corresponding countermeasures could be considered.

Chapter 7

Conclusion and Future Works

This thesis is devoted to design, analysis, and verification of control techniques for safe and secure vehicle platooning.

In Chapter 3, we have focused on security and robustness analysis of vehicle platoons based on a graph-theoretic approach and proposed a novel optimal sensor placement strategy. The vehicles have been assumed to be able to communicate data, such as inter-vehicular distance and speed among each other via wireless communication environments. Both the unidirectional and bidirectional data transfer have been studied. Moreover, the quality of communication links between the vehicles has been considered using edge weights of the underlying path graph topology. The platoon is assumed to be under cyber attacks, and a detector is supposed to choose a strategy to place his monitoring sensors on specific vehicles aiming at increasing the detectability of the attacker. An attacker-detector game has been defined based on which the existence of any possible NE points have been studied. Based on our results, the detector can decide about his sensor placement strategy to increase the security level of the system. Also, robustness analysis of a platoon against adding extra communication links between the vehicles has performed. Furthermore, our study verifies the fact that using a bidirectional communication environment forms a more secure platoon compared to the unidirectional counterpart. Our simulation and experimental results verified the effectiveness of our theoretical analyses. An open avenue for the current research is to extend the underlying graph topology such that it can handle dynamic platoon formations resulted from different vehicle maneuvers such as cut-in/cut-out actions, hence, studying the impacts of those movements on the security of vehicle platooning. Besides, the extension of this work to the dynamic game (with changing network topology) along with generalizing our method for possibly different vehicle dynamics in the platoon referred to as the "heterogeneous" case are left as our future studies.

In Chapter 4, a game-theoretic approach has been proposed to tackle the security challenges of vehicle platoons via a new optimal actuator placement strategy. From the viewpoint of secure platoon control, we studied the problem of threatening a vehicle platoon by one (or more) attacker(s) who tries to deteriorate the platoon control by injection of acceleration attack signal(s) to the longitudinal dynamics of one (or more) of the vehicles. In essence, we focused on the energy needed by the attacker, as our game-payoff, to steer the consensus dynamics of the system towards his desired direction in the state space. In this regard, the attacker(s) basically tries to minimize the amount of energy needed to deviate the system dynamics, while the defender(s) faces this action by attempting to maximize that energy. This confrontation between the attacker(s) and defender(s) was formulated as a Stackelberg game problem, and the algorithm to solve the game was given. Based on the equilibrium point of the game, the defender(s) selects specific nodes(s) to place his actuator(s) in order to mitigate the attack effects as far as possible. Two different scenarios, namely single attacker–single defender and multi attackers–multi defenders were considered. The game formulation, its solution, and simulation results were presented for h–nearest neighbor platoons with different data transfer structures. The proposed technique can be applied to arbitrary data transfer structures employed in different platoon formation topologies. Besides, the effects of increasing the connectivity among the vehicles on the security level of the platoon have been studied. Some experimental tests were also conducted on a real platoon to demonstrate the applicability of the method in practice. An open avenue for this research would be to generalize the work to study a platoon consisting of vehicles with different dynamics referred to as "heterogeneous" platoons. Besides, one can think of a combination of cyber attacks imposed on the platoon; hence, a multi-component game pay-off can be considered reflecting the game values corresponding to different attacks.

Chapter 5 was devoted to a study where we focused on a general static heterogeneous platoon formation under the risk of the widespread so-called DoS attack. A DoS intelligent intruder targets the wireless links by overwhelming the node by invalid requests, hence, either blocks the network or prevents it from timely data transfer. As was explained before, a platoon is mainly consisted of networking and data transmission among the vehicles, forming the cyber layer, and the physical environment composed of the participant cars, forming the physical layer. This cyber-physical system is highly prone to cyber attacks endangering the wireless connectivity among the vehicles. This vulnerability to external attackers needs to be fully addressed as an insecure communication layer in a platoon can cause manipulated and/or missing data received by the followers resulting in dangerous hazards. We proposed a secure control algorithm which enables the platoon to detect and mitigate the devastation imposed by the intruder. The algorithm guarantees the desired

platoon performance in terms of its control objectives together with its safety. Besides, the proposed method tackles the attacker regardless of the employed communication topology among the vehicles. Simulations performed on a TPF heterogeneous platoon with practical vehicle dynamics parameters, indicate the efficacy of the proposed technique. Dealing with other cyber attacks along with achieving additional performance objectives such as driving comfort and ecological driving are left as our future studies. In addition, deriving an upper bound on the attack duration ensuring a safe and secure platooning deserves further research. Furthermore, from a practical point of view, the real-time implementability test via Hardware-in-the-Loop (HIL) can be conducted to assess the turnaround time of the algorithm and potentially optimize it for a faster run. Then, the algorithm can be implemented and tested on a full-scale vehicle platoon.

In Chapter 6, we dealt with a broadly concerned control problem, namely the dynamic heterogeneous platoon control under the risk of a DoS attack. We proposed a Secure–Distributed Nonlinear Model Predictive Control (Secure–DNMPC) framework to ensure a safe and secure dynamic platooning which fulfills both the safe distancing between the cars and speed tracking requirements. The method is capable of handling cut-in/cut-out maneuvers under the premise of the existence of a cyber DoS attack. The algorithm is basically comprised of detection and mitigation phases. Furthermore, we introduced a novel Secure–DNMPC–UKF co-design for the case when the DoS attacker injects a huge amount of time delay in the network compared to the intrinsic practical DSRC time delay. This makes use of the available but outdated data to estimate and predict future states. The proposed approach also provides the opportunity to consider non-ideal sensors which contaminate the measured data. In addition, compromised signals sent through a realistic noisy environment can be considered as well. Simulation results demonstrated the efficacy of the introduced technique. As a future direction, one can think of generalizing the given algorithm to a multi-platooning scenario in which two or more attacked platoons drive in parallel, and arbitrary vehicles wish to exit their own platoon and merge with an adjacent one. Besides, to capture more realistic vehicle dynamics, the tire model can be embedded in the system modeling. Furthermore, generalization of the proposed method to make the controller robust against model parameter uncertainties can be further investigated. Also, other types of attacks and the corresponding countermeasures could be considered.

References

[1] Xue Wang. *Cyber-Physical Systems: A Reference*. Springer-Verlag Berlin Heidelberg, 2021.

[2] Houbing Song, Glenn Fink, and Sabina Jeschke. *Security and Privacy in Cyber-Physical Systems*. Wiley Online Library, 2017.

[3] Alvaro A Cárdenas, Saurabh Amin, and Shankar Sastry. Research challenges for the security of control systems. In *HotSec*, 2008.

[4] Song Guo and Deze Zeng. *Cyber-Physical Systems: Architecture, Security and Application*. Springer, 2019.

[5] Çetin Kaya Koç. *Cyber-Physical Systems Security*. Springer, 2018.

[6] Hao Liu, Ben Niu, and Yuzhe Li. False-data-injection attacks on remote distributed consensus estimation. *IEEE Transactions on Cybernetics*, 2020.

[7] Abdulmalik Humayed, Jingqiang Lin, Fengjun Li, and Bo Luo. Cyber-physical systems securitya survey. *IEEE Internet of Things Journal*, 4(6):1802–1831, 2017.

[8] Yosef Ashibani and Qusay H Mahmoud. Cyber physical systems security: Analysis, challenges and solutions. *Computers & Security*, 68:81–97, 2017.

[9] Saqib Ali, Taiseera Al Balushi, Zia Nadir, and Omar Khadeer Hussain. *Cyber Security for Cyber Physical Systems*, volume 768. Springer, 2018.

[10] Mohammad Hossein Basiri, Nasser L. Azad, and Sebastian Fischmeister. Distributed time-varying kalman filter design and estimation over wireless sensor networks using owa sensor fusion technique. In *28th Mediterranean Conference on Control and Automation (MED), Saint-Raphael, France*. IEEE, 2020.

[11] Ralph Langner. Stuxnet: Dissecting a cyberwarfare weapon. *IEEE Security & Privacy*, 9(3):49–51, 2011.

[12] James P Farwell and Rafal Rohozinski. Stuxnet and the future of cyber war. *Survival*, 53(1):23–40, 2011.

[13] Yao Liu, Peng Ning, and Michael K Reiter. False data injection attacks against state estimation in electric power grids. *ACM Transactions on Information and System Security (TISSEC)*, 14(1):13, 2011.

[14] Yasser Shoukry, Paul Martin, Paulo Tabuada, and Mani Srivastava. Non-invasive spoofing attacks for anti-lock braking systems. In *International Workshop on Cryptographic Hardware and Embedded Systems*, pages 55–72. Springer, 2013.

[15] Dongyao Jia, Kejie Lu, Jianping Wang, Xiang Zhang, and Xuemin Shen. A survey on platoon-based vehicular cyber-physical systems. *IEEE communications surveys & tutorials*, 18(1):263–284, 2016.

[16] Santokh Singh. Critical reasons for crashes investigated in the national motor vehicle crash causation survey. Technical report, 2015.

[17] Transport Canada. Canadian motor vehicle traffic collision statistics. 2014.

[18] *Road Safety in Canada*, 2011. Retrieved from http://www.tc.gc.ca/eng/motorvehiclesafety/tp-tp15145-1201.htm, October, 2018.

[19] Paul Godsmark, B Kirk, V Gill, and B Flemming. Automated vehicles: The coming of the next disruptive technology. 2015.

[20] Retrieved from https://www.oemoffhighway.com/electronics/smart-systems/automated-systems/press-release/20848438/scania-ab-scania-designing-fullscale-autonomous-truck-platooning-operations-in-singapore, November, 2018.

[21] Bart Van Arem, Cornelie JG Van Driel, and Ruben Visser. The impact of cooperative adaptive cruise control on traffic-flow characteristics. *IEEE Transactions on Intelligent Transportation Systems*, 7(4):429–436, 2006.

[22] Feng Gao, Shengbo Eben Li, Yang Zheng, and Dongsuk Kum. Robust control of heterogeneous vehicular platoon with uncertain dynamics and communication delay. *IET Intelligent Transport Systems*, 10(7):503–513, 2016.

[23] Siyuan Gong, Anye Zhou, Jian Wang, Tao Li, and Srinivas Peeta. Cooperative adaptive cruise control for a platoon of connected and autonomous vehicles considering dynamic information flow topology. *arXiv preprint arXiv:1807.02224*, 2018.

[24] Lei Lin, Siyuan Gong, and Tao Li. Deep learning-based human-driven vehicle trajectory prediction and its application for platoon control of connected and autonomous vehicles. In *The Autonomous Vehicles Symposium*, 2018.

[25] Xiaotian Sun, Roberto Horowitz, and Chin-Woo Tan. An efficient lane change maneuver for platoons of vehicles in an automated highway system. In *ASME 2003 International Mechanical Engineering Congress and Exposition*, pages 355–362. American Society of Mechanical Engineers, 2003.

[26] Stanley Lam and Jayantha Katupitiya. Cooperative autonomous platoon maneuvers on highways. In *Advanced Intelligent Mechatronics (AIM), 2013 IEEE/ASME International Conference on*, pages 1152–1157. IEEE, 2013.

[27] Vicente Milanés and Steven E Shladover. Handling cut-in vehicles in strings of cooperative adaptive cruise control vehicles. *Journal of Intelligent Transportation Systems*, 20(2):178–191, 2016.

[28] Vicente Milanés, Steven E Shladover, John Spring, Christopher Nowakowski, Hiroshi Kawazoe, and Masahide Nakamura. Cooperative adaptive cruise control in real traffic situations. *IEEE Trans. Intelligent Transportation Systems*, 15(1):296–305, 2014.

[29] Ziran Wang, BaekGyu Kim, Hiromitsu Kobayashi, Guoyuan Wu, and Matthew J Barth. Agent-based modeling and simulation of connected and automated vehicles using game engine: A cooperative on-ramp merging study. *arXiv preprint arXiv:1810.09952*, 2018.

[30] Elham Semsar Kazerooni and Jeroen Ploeg. Interaction protocols for cooperative merging and lane reduction scenarios. In *Intelligent Transportation Systems (ITSC), 2015 IEEE 18th International Conference on*, pages 1964–1970. IEEE, 2015.

[31] Hoai Hoang Bengtsson, Lei Chen, Alexey Voronov, and Cristofer Englund. Interaction protocol for highway platoon merge. In *Intelligent Transportation Systems (ITSC), 2015 IEEE 18th International Conference on*, pages 1971–1976. IEEE, 2015.

[32] Takeshi Sakaguchi, Atsuya Uno, and Sadayuki Tsugawa. Inter-vehicle communications for merging control. In *Vehicle Electronics Conference, 1999.(IVEC'99) Proceedings of the IEEE International*, pages 365–370. IEEE, 1999.

[33] Michael P Vitus and Claire J Tomlin. A hybrid method for chance constrained control in uncertain environments. In *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*, pages 2177–2182. IEEE, 2012.

[34] Michael P Vitus and Claire J Tomlin. A probabilistic approach to planning and control in autonomous urban driving. In *Decision and Control (CDC), 2013 IEEE 52nd Annual Conference on*, pages 2459–2464. IEEE, 2013.

[35] Xiangjun Qian, Jean Gregoire, Arnaud De La Fortelle, and Fabien Moutarde. Decentralized model predictive control for smooth coordination of automated vehicles at intersection. In *Control Conference (ECC), 2015 European*, pages 3452–3458. IEEE, 2015.

[36] Laleh Makarem and Denis Gillet. Model predictive coordination of autonomous vehicles crossing intersections. In *Intelligent Transportation Systems-(ITSC), 2013 16th International IEEE Conference on*, pages 1799–1804. IEEE, 2013.

[37] David Lenz, Tobias Kessler, and Alois Knoll. Stochastic model predictive controller with chance constraints for comfortable and safe driving behavior of autonomous vehicles. In *Intelligent Vehicles Symposium*, pages 292–297, 2015.

[38] Dominik Moser, Luigi del Re, and Stephen Jones. A risk constrained control approach for adaptive cruise control. In *Control Technology and Applications (CCTA), 2017 IEEE Conference on*, pages 578–583. IEEE, 2017.

[39] Ashwin Carvalho, Yiqi Gao, Stéphanie Lefevre, and Francesco Borrelli. Stochastic predictive control of autonomous vehicles in uncertain environments. In *12th International Symposium on Advanced Vehicle Control*, pages 712–719, 2014.

[40] Ashwin Mark Carvalho. *Predictive Control under Uncertainty for Safe Autonomous Driving: Integrating Data-Driven Forecasts with Control Design*. PhD thesis, UC Berkeley, 2016.

[41] Muhammad Awais Javed and Elyes Ben Hamida. On the interrelation of security, qos, and safety in cooperative its. *IEEE Transactions on Intelligent Transportation Systems*, 18(7):1943–1957, 2017.

[42] Ulrich Lang and Rudolf Schreiner. Managing security in intelligent transport systems. In *Intelligent Transportation Systems (ITSC), 2015 IEEE 18th International Conference on*, pages 48–53. IEEE, 2015.

[43] Philip Koopman and Michael Wagner. Autonomous vehicle safety: An interdisciplinary challenge. *IEEE Intelligent Transportation Systems Magazine*, 9(1):90–96, 2017.

[44] Ke Huang, Xianfeng Yang, Yang Lu, Chunting Chris Mi, and Prathyusha Kondlapudi. Ecological driving system for connected/automated vehicles using a two-stage control hierarchy. *IEEE Transactions on Intelligent Transportation Systems*, 2018.

[45] Ge Guo and Qiong Wang. Fuel-efficient en route speed planning and tracking control of truck platoons. *IEEE Transactions on Intelligent Transportation Systems*, 2018.

[46] Tao Zhang, Helder Antunes, and Siddhartha Aggarwal. Defending connected vehicles against malware: Challenges and a solution framework. *IEEE Internet of Things journal*, 1(1):10–21, 2014.

[47] Shengbo Eben Li, Yang Zheng, Keqiang Li, Yujia Wu, J Karl Hedrick, Feng Gao, and Hongwei Zhang. Dynamical modeling and distributed control of connected and automated vehicles: Challenges and opportunities. *IEEE Intelligent Transportation Systems Magazine*, 9(3):46–58, 2017.

[48] Steven E Shladover. Cooperative (rather than autonomous) vehicle-highway automation systems. *IEEE Intelligent Transportation Systems Magazine*, 1(1):10–19, 2009.

[49] Derek Caveney. Cooperative vehicular safety applications. *IEEE Control Systems Magazine*, 30(4):38–53, 2010.

[50] Darbha Swaroop, J Karl Hedrick, and Seibum B Choi. Direct adaptive longitudinal control of vehicle platoons. *IEEE Transactions on Vehicular Technology*, 50(1):150–161, 2001.

[51] Kakan C Dey, Li Yan, Xujie Wang, Yue Wang, Haiying Shen, Mashrur Chowdhury, Lei Yu, Chenxi Qiu, and Vivekgautham Soundararaj. A review of communication, driver characteristics, and controls aspects of cooperative adaptive cruise control (cacc). *IEEE Transactions on Intelligent Transportation Systems*, 17(2):491–509, 2016.

[52] Joshua E Siegel, Dylan C Erb, and Sanjay E Sarma. A survey of the connected vehicle landscape-architectures, enabling technologies, applications, and development areas. *IEEE Transactions on Intelligent Transportation Systems*, 19(8):2391–2406, 2018.

[53] Vicente Milanés, Steven E Shladover, John Spring, Christopher Nowakowski, Hiroshi Kawazoe, and Masahide Nakamura. Cooperative adaptive cruise control in real traffic situations. *IEEE Transactions on Intelligent Transportation Systems*, 15(1):296–305, 2013.

[54] Levent Guvenc, Ismail Meriç Can Uygan, Kerim Kahraman, Raif Karaahmetoglu, Ilker Altay, Mutlu Senturk, Mümin Tolga Emirler, Ahu Ece Hartavi Karci, Bilin Aksun Guvenc, Erdinç Altug, et al. Cooperative adaptive cruise control implementation of team mekar at the grand cooperative driving challenge. *IEEE Transactions on Intelligent Transportation Systems*, 13(3):1062–1074, 2012.

[55] Kristoffer Lidström, Katrin Sjöberg, Ulf Holmberg, Johan Andersson, Fredrik Bergh, Mattias Bjäde, and Spencer Mak. A modular cacc system integration and design. *IEEE Transactions on Intelligent Transportation Systems*, 13(3):1050, 2012.

[56] Bijan Sakhdari and Nasser L Azad. Adaptive tube-based nonlinear mpc for economic autonomous cruise control of plug-in hybrid electric vehicles. *IEEE Transactions on Vehicular Technology*, 67(12):11390–11401, 2018.

[57] Bijan Sakhdari and Nasser L Azad. A distributed reference governor approach to ecological cooperative adaptive cruise control. *IEEE Transactions on Intelligent Transportation Systems*, 19(5):1496–1507, 2018.

[58] Christopher Nowakowski, Steven E Shladover, Xiao-Yun Lu, Deborah Thompson, and Aravind Kailas. *Cooperative adaptive cruise control (CACC) for truck platooning: Operational concept alternatives*, 2015.

[59] Ziran Wang, Guoyuan Wu, and Matthew J Barth. Developing a distributed consensus-based cooperative adaptive cruise control system for heterogeneous vehicles with predecessor following topology. *Journal of Advanced Transportation*, 2017, 2017.

[60] Chaojie Wang, Siyuan Gong, Anye Zhou, Tao Li, and Srinivas Peeta. Cooperative adaptive cruise control for connected autonomous vehicles by factoring communication-related constraints. *Transportation Research Part C: Emerging Technologies*, 2019.

[61] Yang Zheng, Shengbo Eben Li, Keqiang Li, Francesco Borrelli, and J Karl Hedrick. Distributed model predictive control for heterogeneous vehicle platoons under unidirectional topologies. *IEEE Transactions on Control Systems Technology*, 25(3):899–910, 2017.

[62] Srdjan S Stankovic, Milorad J Stanojevic, and Dragoslav D Siljak. Decentralized overlapping control of a platoon of vehicles. *IEEE Transactions on Control Systems Technology*, 8(5):816–832, 2000.

[63] Ellen van Nunen, Joey Reinders, Elham Semsar-Kazerooni, and Nathan Van De Wouw. String stable model predictive cooperative adaptive cruise control for heterogeneous platoons. *IEEE Transactions on Intelligent Vehicles*, 4(2):186–196, 2019.

[64] Francesco Acciani, Paolo Frasca, Geert Heijenk, and Anton Stoorvogel. Stochastic string stability of vehicle platoons via cooperative adaptive cruise control with lossy communication. *arXiv preprint arXiv:1905.04779*, 2019.

[65] Feng Gao, Xiaosong Hu, Shengbo Eben Li, Keqiang Li, and Qi Sun. Distributed adaptive sliding mode control of vehicular platoon with uncertain interaction topology. *IEEE Transactions on Industrial Electronics*, 65(8):6352–6361, 2018.

[66] Yongfu Li, Chuancong Tang, Srinivas Peeta, and Yibing Wang. Nonlinear consensus-based connected vehicle platoon control incorporating car-following interactions and heterogeneous time delays. *IEEE Transactions on Intelligent Transportation Systems*, 2018.

[67] Peng Liu, Arda Kurt, and Umit Ozguner. Distributed model predictive control for cooperative and flexible vehicle platooning. *IEEE Transactions on Control Systems Technology*, (99):1–14, 2018.

[68] Yang Zheng, Shengbo Eben Li, Jianqiang Wang, Dongpu Cao, and Keqiang Li. Stability and scalability of homogeneous vehicular platoon: Study on the influence of information flow topologies. *IEEE Transactions on Intelligent Transportation Systems*, 17(1):14–26, 2016.

[69] Mohammad Pirani, Ehsan Hashemi, John W Simpson-Porco, Baris Fidan, and Amir Khajepour. Graph theoretic approach to the robustness of k-nearest neighbor vehicle platoons. *IEEE Transactions on Intelligent Transportation Systems*, 18(11):3218–3224, 2017.

[70] C Lei, EM Van Eenennaam, W Klein Wolterink, G Karagiannis, Geert Heijenk, and J Ploeg. Impact of packet loss on cacc string stability performance. In *ITS Telecommunications (ITST), 2011 11th International Conference on*, pages 381–386. IEEE, 2011.

[71] Jeroen Ploeg, Elham Semsar-Kazerooni, Guido Lijster, Nathan van de Wouw, and Henk Nijmeijer. Graceful degradation of cacc performance subject to unreliable wireless communication. In *Intelligent Transportation Systems-(ITSC), 2013 16th International IEEE Conference on*, pages 1210–1216. IEEE, 2013.

[72] Fengzhong Qu, Fei-Yue Wang, and Liuqing Yang. Intelligent transportation spaces: vehicles, traffic, communications, and beyond. *IEEE Communications Magazine*, 48(11), 2010.

[73] Theodore L Willke, Patcharinee Tientrakool, and Nicholas F Maxemchuk. A survey of inter-vehicle communication protocols and their applications. *IEEE Communications Surveys & Tutorials*, 11(2), 2009.

[74] Jinhua Guo and Nathan Balon. Vehicular ad hoc networks and dedicated short-range communication. *University of Michigan*, 2006.

[75] Xuehai Xiang, Wenhu Qin, and Binfu Xiang. Research on a DSRC-based rear-end collision warning model. *IEEE Transactions on Intelligent Transportation Systems*, 15(3):1054–1065, 2014.

[76] ASTM International. *Standard Specification for Telecommunications and Information Exchange Between Roadside and Vehicle Systems – 5-GHz Band Dedicated Short-Range Communications (DSRC), Medium Access Control (MAC), and Physical Layer (PHY) Specifications, West Conshohocken, PA, www.astm.org*, 2018.

[77] John B Kenney. Dedicated short-range communications (dsrc) standards in the united states. *Proceedings of the IEEE*, 99(7):1162–1182, 2011.

[78] Mani Amoozadeh, Arun Raghuramu, Chen-Nee Chuah, Dipak Ghosal, H Michael Zhang, Jeff Rowe, and Karl Levitt. Security vulnerabilities of connected vehicle streams and their impact on cooperative driving. *IEEE Communications Magazine*, 53(6):126–132, 2015.

[79] Soodeh Dadras and Chris Winstead. *Insider Vs. Outsider Threats to Autonomous Vehicle Platooning*, 2018. [Online]. Available: https://digitalcommons.usu.edu/cgi/viewcontent.cgi?article=1773&context=researchweek.

[80] Maxim Raya and Jean-Pierre Hubaux. Securing vehicular ad hoc networks. *Journal of computer security*, 15(1):39–68, 2007.

[81] Frank Kargl, Panagiotis Papadimitratos, Levente Buttyan, Michael Müter, Elmar Schoch, Bjorn Wiedersheim, Ta-Vinh Thong, Giorgio Calandriello, Albert Held, Antonio Kung, et al. Secure vehicular communication systems: implementation, performance, and research challenges. *IEEE Communications magazine*, 46(11):110–118, 2008.

[82] Ryan M Gerdes, Chris Winstead, and Kevin Heaslip. Cps: an efficiency-motivated attack against autonomous vehicular transportation. In *Proceedings of the 29th Annual Computer Security Applications Conference*, pages 99–108. ACM, 2013.

[83] Maria Azees, Pandi Vijayakumar, and Lazarus Jegatha Deborah. Comprehensive survey on security services in vehicular ad-hoc networks. *IET Intelligent Transport Systems*, 10(6):379–388, 2016.

[84] André Teixeira, Iman Shames, Henrik Sandberg, and Karl Henrik Johansson. A secure control framework for resource-limited adversaries. *Automatica*, 51:135–148, 2015.

[85] Jezdimir Miloševič, Takashi Tanaka, Henrik Sandberg, and Karl Henrik Johansson. Analysis and mitigation of bias injection attacks against a kalman filter. *IFAC-PapersOnLine*, 50(1):8393–8398, 2017.

[86] Soodeh Dadras, Ryan M Gerdes, and Rajnikant Sharma. Vehicular platooning in an adversarial environment. In *Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security*, pages 167–178. ACM, 2015.

[87] Bruce DeBruhl, Sean Weerakkody, Bruno Sinopoli, and Patrick Tague. Is your commute driving you crazy?: a study of misbehavior in vehicular platoons. In *Proceedings of the 8th ACM Conference on Security & Privacy in Wireless and Mobile Networks*, page 22. ACM, 2015.

[88] Daniel D Dunn. *Attacker-induced traffic flow instability in a stream of automated vehicles*. Utah State University, 2015.

[89] Daniel D Dunn, Samuel A Mitchell, Imran Sajjad, Ryan M Gerdes, Rajnikant Sharma, and Ming Li. Regular: Attacker-induced traffic flow instability in a stream of semi-automated vehicles. In *2017 47th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, pages 499–510. IEEE, 2017.

[90] Bidisha Biswas. Analysis of false data injection in vehicle platooning. Master's thesis, Utah State University, 2014.

[91] *CAN bus connecting different internal vehicle components*, 2020. [Online]. Available: https://www.enter.co/cultura-digital/autotecnologia/el-99-de-los-carros-modernos-son-vulnerables-a-ser-hackeados/.

[92] *A Brief History of Car Hacking 2010 to the Present*, 2017. [Online]. Available: https://smart.gi-de.com/2017/08/brief-history-car-hacking-2010-present/.

[93] Tamás Bécsi, Szilárd Aradi, and Péter Gáspár. Security issues and vulnerabilities in connected car systems. In *Models and Technologies for Intelligent Transportation Systems (MT-ITS), 2015 International Conference on*, pages 477–482. IEEE, 2015.

[94] *Beemer, Open Thyself! – Security vulnerabilities in BMW's ConnectedDrive*, 2015. [Online]. Available: https://www.heise.de/ct/artikel/Beemer-Open-Thyself-Security-vulnerabilities-in-BMW-s-ConnectedDrive-2540957.html.

[95] Harvey J Miller and Shih-Lung Shaw. *Geographic information systems for transportation: principles and applications*. Oxford University Press on Demand, 2001.

[96] Xiaodong Lin, Xiaoting Sun, Pin-Han Ho, and Xuemin Shen. Gsis: A secure and privacy-preserving protocol for vehicular communications. *IEEE Transactions on vehicular technology*, 56(6):3442–3456, 2007.

[97] Panagiotis Papadimitratos, Levente Buttyan, Jean-Pierre Hubaux, Frank Kargl, Antonio Kung, and Maxim Raya. Architecture for secure and private vehicular communications. In *Telecommunications, 2007. ITST'07. 7th International Conference on ITS*, pages 1–6. IEEE, 2007.

[98] Roberto Merco, Zoleikha Abdollahi Biron, and Pierluigi Pisu. Replay attack detection in a platoon of connected vehicles with cooperative adaptive cruise control. In *2018 Annual American Control Conference (ACC)*, pages 5582–5587. IEEE, 2018.

[99] Lian Cui, Jia Hu, B Brian Park, and Pavle Bujanovic. Development of a simulation platform for safety impact analysis considering vehicle dynamics, sensor errors, and communication latencies: Assessing cooperative adaptive cruise control under cyber attack. *Transportation Research Part C: Emerging Technologies*, 97:1–22, 2018.

[100] Eman Mousavinejad, Fuwen Yang, Qing-Long Han, Quanwei Qiu, and Ljubo Vlacic. Cyber attack detection in platoon-based vehicular networked control systems. In *2018 IEEE 27th International Symposium on Industrial Electronics (ISIE)*, pages 603–608. IEEE, 2018.

[101] Imran Sajjad, Daniel D Dunn, Rajnikant Sharma, and Ryan Gerdes. Attack mitigation in adversarial platooning using detection-based sliding mode control. In *Proceedings of the First ACM Workshop on Cyber-Physical Systems-Security and/or PrivaCy*, pages 43–53. ACM, 2015.

[102] Raj Gautam Dutta, Feng Yu, Teng Zhang, Yaodan Hu, and Yier Jin. Security for safety: a path toward building trusted autonomous vehicles. In *Proceedings of the International Conference on Computer-Aided Design*, page 92. ACM, 2018.

[103] Nikita Lyamin, Alexey Vinel, Magnus Jonsson, and Jonathan Loo. Real-time detection of denial-of-service attacks in ieee 802.11 p vehicular networks. *IEEE Communications letters*, 18(1):110–113, 2014.

[104] Hamza Fawzi, Paulo Tabuada, and Suhas Diggavi. Security for control systems under sensor and actuator attacks. In *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*, pages 3412–3417. Citeseer, 2012.

[105] Saurabh Amin, Alvaro A Cárdenas, and S Shankar Sastry. Safe and secure networked control systems under denial-of-service attacks. In *International Workshop on Hybrid Systems: Computation and Control*, pages 31–45. Springer, 2009.

[106] Yilin Mo and Bruno Sinopoli. Secure control against replay attacks. In *Communication, Control, and Computing, 2009. Allerton 2009. 47th Annual Allerton Conference on*, pages 911–918. IEEE, 2009.

[107] Yilin Mo, Emanuele Garone, Alessandro Casavola, and Bruno Sinopoli. False data injection attacks against state estimation in wireless sensor networks. In *Decision and Control (CDC), 2010 49th IEEE Conference on*, pages 5967–5972. IEEE, 2010.

[108] Rohan Chabukswar, Bruno Sinopoli, Gabor Karsai, Annarita Giani, Himanshu Neema, and Andrew Davis. Simulation of network attacks on scada systems. In *First Workshop on Secure Control Systems*, pages 587–592, 2010.

[109] Mohammad Hossein Basiri, John G Thistle, John W Simpson-Porco, and Sebastian Fischmeister. Kalman filter based secure state estimation and individual attacked sensor detection in cyber-physical systems. In *2019 American Control Conference (ACC), Philadelphia, USA*, pages 3841–3848. IEEE, 2019.

[110] Q. Zhu and T. Basar. Game-theoretic methods for robustness, security, and resilience of cyberphysical control systems: Games-in-games principle for optimal cross-layer resilient control systems. In *IEEE control systems*, volume 35, pages 45–65, 2015.

[111] Yuzhe Li, Ling Shi, Peng Cheng, Jiming Chen, and Daniel E Quevedo. Jamming attacks on remote state estimation in cyber-physical systems: A game-theoretic approach. *IEEE Transactions on Automatic Control*, 60(10):2831–2836, 2015.

[112] A. Gupta, C. Langbort, and T. Basar. Optimal control in the presence of an intelligent jammer with limited actions. *49th IEEE Conference on Decision and Control*, pages 1096–1101, 2010.

[113] J. P. Hubaux M. Felegyhazi. *Game Theory in Wireless Networks: A Tutorial*. EPFL Technical report, 2006.

[114] Jason R Marden, Gürdal Arslan, and Jeff S Shamma. Cooperative control and potential games. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 39(6):1393–1407, 2009.

[115] M. Pirani, E. Nekouie, H. Sandberg, and K.H.Johansson. Attacker-detector visibility game with applications to vehicular platooning. *arXiv: 1809.08331v1*, 2018.

[116] P. N. Brown and H. Borowski ND J. R Marden. *Security Against Impersonation Attacks in Distributed Systems*. arXiv preprint arXiv:1711.00609, 2017.

[117] S. Amin, G. A. Schwartz, and S. S. Sastry. Security of interdependent and identical networked control systems. *Automatica*, pages 186–192, 2013.

[118] Wellington Lobato, Denis Rosario, Mario Gerla, and Leandro A Villas. Platoon-based driving protocol based on game theory for multimedia transmission over vanet. In *GLOBECOM 2017-2017 IEEE Global Communications Conference*, pages 1–6. IEEE, 2017.

[119] Zhiheng Xu and Quanyan Zhu. A game-theoretic approach to secure control of communication-based train control systems under jamming attacks. In *Proceedings of the 1st International Workshop on Safe Control of Connected and Autonomous Vehicles*, pages 27–34. ACM, 2017.

[120] Youssef Abou Harfouch, Shuai Yuan, and Simone Baldi. An adaptive switched control approach to heterogeneous platooning with intervehicle communication losses. *IEEE Transactions on Control of Network Systems*, 5(3):1434–1444, 2017.

[121] Victor S Dolk, Jeroen Ploeg, and WP Maurice H Heemels. Event-triggered control for string-stable vehicle platooning. *IEEE Transactions on Intelligent Transportation Systems*, 18(12):3486–3500, 2017.

[122] Sinan Öncü, Jeroen Ploeg, Nathan Van de Wouw, and Henk Nijmeijer. Cooperative adaptive cruise control: Network-aware analysis of string stability. *IEEE Transactions on Intelligent Transportation Systems*, 15(4):1527–1537, 2014.

[123] Alvaro Cardenas, Saurabh Amin, Bruno Sinopoli, Annarita Giani, Adrian Perrig, Shankar Sastry, et al. Challenges for securing cyber physical systems. In *Workshop on future directions in cyber-physical systems security*, volume 5, 2009.

[124] Hamza Fawzi, Paulo Tabuada, and Suhas Diggavi. Secure estimation and control for cyber-physical systems under adversarial attacks. *IEEE Transactions on Automatic control*, 59(6):1454–1467, 2014.

[125] Alvaro A Cardenas, Saurabh Amin, and Shankar Sastry. Secure control: Towards survivable cyber-physical systems. In *2008 The 28th International Conference on Distributed Computing Systems Workshops*, pages 495–500. IEEE, 2008.

[126] Fabio Pasqualetti, Florian Dörfler, and Francesco Bullo. Attack detection and identification in cyber-physical systems. *IEEE Transactions on Automatic Control*, 58(11):2715–2729, 2013.

[127] Yin-Chen Liu, Gianluca Bianchin, and Fabio Pasqualetti. Secure trajectory planning against undetectable spoofing attacks. *Automatica*, 112:108655, 2020.

[128] Sean Weerakkody, Xiaofei Liu, Sang Hyuk Son, and Bruno Sinopoli. A graph-theoretic characterization of perfect attackability for secure design of distributed control systems. *IEEE Transactions on Control of Network Systems*, 4(1):60–70, 2016.

[129] Zoleikha Abdollahi Biron, Satadru Dey, and Pierluigi Pisu. Real-time detection and estimation of denial of service attack in connected vehicle systems. *IEEE Transactions on Intelligent Transportation Systems*, (99):1–10, 2018.

[130] Wubing B Qin and Gábor Orosz. Experimental validation of string stability for connected vehicles subject to information delay. *IEEE Transactions on Control Systems Technology*, 2019.

[131] Wubing B Qin, Marcella M Gomez, and Gábor Orosz. Stability analysis of connected cruise control with stochastic delays. In *2014 American Control Conference*, pages 4624–4629. IEEE, 2014.

[132] Mohammad Hossein Basiri, Mohammad Pirani, Nasser L Azad, and Sebastian Fischmeister. Security of vehicle platooning: A game-theoretic approach. *IEEE Access*, 7(1):185565–185579, 2019.

[133] Danda B Rawat and Chandra Bajracharya. Vehicular cyber physical systems. Technical report, Springer, 2017.

[134] Simon Parkinson, Paul Ward, Kyle Wilson, and Jonathan Miller. Cyber threats facing autonomous and connected vehicles: Future challenges. *IEEE transactions on intelligent transportation systems*, 18(11):2898–2915, 2017.

[135] Jonathan Petit and Steven E Shladover. Potential cyberattacks on automated vehicles. *IEEE Transactions on Intelligent transportation systems*, 16(2):546–556, 2014.

[136] Christine Laurendeau and Michel Barbeau. Threats to security in dsrc/wave. In *International Conference on Ad-Hoc Networks and Wireless*, pages 266–279. Springer, 2006.

[137] Samuel Woo, Hyo Jin Jo, and Dong Hoon Lee. A practical wireless attack on the connected car and security protocol for in-vehicle can. *IEEE Transactions on intelligent transportation systems*, 16(2):993–1006, 2015.

[138] Pierre Kleberger, Tomas Olovsson, and Erland Jonsson. Security aspects of the in-vehicle network in the connected car. In *2011 IEEE Intelligent Vehicles Symposium (IV)*, pages 528–533. IEEE, 2011.

[139] Mohammad Pirani, Ehsan Hashemi, Baris Fidan, John W Simpson-Porco, Henrik Sandberg, and Karl Henrik Johansson. Resilient estimation and control on k-nearest neighbor platoons: A network-theoretic approach. *IFAC-PapersOnLine*, 51(23):22–27, 2018.

[140] Khaled Rabieh, Mohamed MEA Mahmoud, and Mohamed Younis. Privacy-preserving route reporting schemes for traffic management systems. *IEEE Transactions on Vehicular Technology*, 66(3):2703–2713, 2017.

[141] Yuheng Du, Mashrur Chowdhury, Mizanur Rahman, Kakan Dey, Amy Apon, Andre Luckow, and Linh Bao Ngo. A distributed message delivery infrastructure for connected vehicle technology applications. *IEEE Transactions on Intelligent Transportation Systems*, 19(3):787–801, 2018.

[142] Elham Semsar-Kazerooni and Jeroen Ploeg. Performance analysis of a cooperative adaptive cruise controller subject to dynamic time headway. In *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, pages 1190–1195. IEEE, 2013.

[143] Zoleikha Abdollahi Biron, Satadru Dey, and Pierluigi Pisu. Sensor fault diagnosis of connected vehicles under imperfect communication network. In *ASME, Dynamic Systems and Control Conference*, volume 1, page V001T16A003, 2016.

[144] Quanyan Zhu and Tamer Basar. Game-theoretic methods for robustness, security, and resilience of cyberphysical control systems: games-in-games principle for optimal cross-layer resilient control systems. *IEEE Control Systems Magazine*, 35(1):46–65, 2015.

[145] Abhishek Gupta, Cédric Langbort, and Tamer Başar. Optimal control in the presence of an intelligent jammer with limited actions. In *49th IEEE Conference on Decision and Control (CDC)*, pages 1096–1101. IEEE, 2010.

[146] Ather Gattami, Assad Al Alam, Karl H Johansson, and Claire J Tomlin. Establishing safety for heavy duty vehicle platooning: A game theoretical approach. *IFAC Proceedings Volumes*, 44(1):3818–3823, 2011.

[147] Mohammad Hossein Manshaei, Quanyan Zhu, Tansu Alpcan, Tamer Bacşar, and Jean-Pierre Hubaux. Game theory meets network security and privacy. *ACM Computing Surveys (CSUR)*, 45(3):25, 2013.

[148] Saurabh Amin, Galina A Schwartz, and S Shankar Sastry. Security of interdependent and identical networked control systems. *Automatica*, 49(1):186–192, 2013.

[149] Philip N Brown, Holly Borowski, and Jason R Marden. Security against impersonation attacks in distributed systems. *IEEE Transactions on Control of Network Systems*, 2018.

[150] Fan Bai, Daniel D Stancil, and Hariharan Krishnan. Toward understanding characteristics of dedicated short range communications (DSRC) from a perspective of vehicular network engineers. In *Proceedings of the sixteenth annual international conference on Mobile computing and networking*, pages 329–340. ACM, 2010.

[151] Donghoon Shin, Kangmun Park, and Manbok Park. Effects of vehicular communication on risk assessment in automated driving vehicles. *Applied Sciences*, 8(12):2632, 2018.

[152] V Shivaldova, G Maier, D Smely, N Czink, A Paier, and CF Mecklenbräuker. Performance analysis of vehicle-to-vehicle tunnel measurements at 5.9 ghz. In *2011 URSI General Assembly and Scientific Symposium*, pages 1–4. IEEE, 2011.

[153] Xinzhou Wu, Sundar Subramanian, Ratul Guha, Robert G White, Junyi Li, Kevin W Lu, Anthony Bucceri, and Tao Zhang. Vehicular communications using DSRC: challenges, enhancements, and evolution. *IEEE Journal on Selected Areas in Communications*, 31(9):399–408, 2013.

[154] He Hao and Prabir Barooah. Stability and robustness of large platoons of vehicles with double-integrator models and nearest neighbor interaction. *International Journal of Robust and Nonlinear Control*, 23(18):2097–2122, 2013.

[155] He Hao, Prabir Barooah, and JJP Veerman. Effect of network structure on the stability margin of large vehicle formation with distributed control. In *Decision and Control (CDC), 2010 49th IEEE Conference on*, pages 4783–4788. IEEE, 2010.

[156] Stephen Boyd, Laurent El Ghaoui, Eric Feron, and Venkataramanan Balakrishnan. *Linear matrix inequalities in system and control theory*, volume 15. Siam, 1994.

[157] Shreyas Sundaram and Christoforos N Hadjicostis. Distributed function calculation via linear iterative strategies in the presence of malicious agents. *IEEE Transactions on Automatic Control*, 56(7):1495–1508, 2010.

[158] Mohammad Pirani and Shreyas Sundaram. On the smallest eigenvalue of grounded laplacian matrices. *IEEE Transactions on Automatic Control*, 61(2):509–514, 2015.

[159] P. V. Mieghem. *Graph Spectra for Complex Networks*. Cambridge University Press, 2011.

[160] D Swaroop and J Karl Hedrick. String stability of interconnected systems. *IEEE transactions on automatic control*, 41(3):349–357, 1996.

[161] Fabio Pasqualetti, Sandro Zampieri, and Francesco Bullo. Controllability metrics, limitations and algorithms for complex networks. *IEEE Transactions on Control of Network Systems*, 1(1):40–52, 2014.

[162] Tyler H Summers, Fabrizio L Cortesi, and John Lygeros. On submodularity and controllability in complex dynamical networks. *IEEE Transactions on Control of Network Systems*, 3(1):91–101, 2016.

[163] Mohammad Hossein Basiri, Mohammad Pirani, Nasser L Azad, and Sebastian Fischmeister. Security-aware optimal actuator placement in vehicle platooning. *Asian Journal of Control*, 2020.

[164] Yang Zheng, Shengbo Eben Li, Keqiang Li, and Le-Yi Wang. Stability margin improvement of vehicular platoon considering undirected topology and asymmetric control. *IEEE Transactions on Control Systems Technology*, 24(4):1253–1265, 2016.

[165] DVAHG Swaroop and J Karl Hedrick. Constant spacing strategies for platooning in automated highway systems. *Journal of dynamic systems, measurement, and control*, 121(3):462–470, 1999.

[166] Lingyun Xiao and Feng Gao. Practical string stability of platoon of adaptive cruise control vehicles. *IEEE Transactions on intelligent transportation systems*, 12(4):1184–1194, 2011.

[167] Seyed Mehran Dibaji and Hideaki Ishii. Resilient consensus of second-order agent networks: Asynchronous update rules with delays. *Automatica*, 81:123–132, 2017.

[168] Emma Tegling and Henrik Sandberg. On the coherence of large-scale networks with distributed pi and pd control. *IEEE control systems letters*, 1(1):170–175, 2017.

[169] Wei Ren, Randal W Beard, and Ella M Atkins. Information consensus in multivehicle cooperative control. *IEEE Control systems magazine*, 27(2):71–82, 2007.

[170] Richard H. Bartels and George W Stewart. Solution of the matrix equation $ax+xb = c$. *Communications of the ACM*, 15(9):820–826, 1972.

[171] Sven J Hammarling. Numerical solution of the stable, non-negative definite Lyapunov equation lyapunov equation. *IMA Journal of Numerical Analysis*, 2(3):303–323, 1982.

[172] Jing-Rebecca Li and Jacob White. Low rank solution of Lyapunov equations. *SIAM Journal on Matrix Analysis and Applications*, 24(1):260–280, 2002.

[173] Jeroen Ploeg, Nathan Van De Wouw, and Henk Nijmeijer. Lp string stability of cascaded systems: Application to vehicle platooning. *IEEE Transactions on Control Systems Technology*, 22(2):786–793, 2013.

[174] Men Long, Chwan-Hwa Wu, and John Y Hung. Denial of service attacks on network-based control systems: impact and mitigation. *IEEE Transactions on Industrial Informatics*, 1(2):85–96, 2005.

[175] Nikita Lyamin, Alexey Vinel, Magnus Jonsson, and Jonathan Loo. Real-time detection of denial-of-service attacks in ieee 802.11 p vehicular networks. *IEEE Communications letters*, 18(1):110–113, 2013.

[176] Zoliekha Abdollahi Biron, Satadru Dey, and Pierluigi Pisu. Resilient control strategy under denial of service in connected vehicles. In *2017 American Control Conference (ACC)*, pages 4971–4976. IEEE, 2017.

[177] Mohammad Hossein Basiri, Nasser L. Azad, and Sebastian Fischmeister. Attack resilient heterogeneous vehicle platooning using secure distributed nonlinear model predictive control. In *28th Mediterranean Conference on Control and Automation (MED), Saint-Raphael, France*. IEEE, 2020.

[178] Frank L Lewis, Hongwei Zhang, Kristian Hengster-Movric, and Abhijit Das. *Cooperative control of multi-agent systems: optimal and adaptive design approaches*. Springer Science & Business Media, 2013.

[179] Yuan Yuan, Quanyan Zhu, Fuchun Sun, Qinyi Wang, and Tamer Başar. Resilient control of cyber-physical systems against denial-of-service attacks. In *2013 6th International Symposium on Resilient Control Systems (ISRCS)*, pages 54–59. IEEE, 2013.

[180] Heng Zhang, Peng Cheng, Ling Shi, and Jiming Chen. Optimal denial-of-service attack scheduling with energy constraint. *IEEE Transactions on Automatic Control*, 60(11):3023–3028, 2015.

[181] Qi Sun, Kunwu Zhang, and Yang Shi. Resilient model predictive control of cyber-physical systems under dos attacks. *IEEE Transactions on Industrial Informatics*, 2019.

[182] Roger A Horn and Charles R Johnson. *Matrix analysis*. Cambridge university press, 2012.

[183] TamáS Keviczky, Francesco Borrelli, and Gary J Balas. Decentralized receding horizon control for large scale dynamically decoupled systems. *Automatica*, 42(12):2105–2115, 2006.

[184] Jian-Qiang Wang, Shengbo Eben Li, Yang Zheng, and Xiao-Yun Lu. Longitudinal collision mitigation via coordinated braking of multiple vehicles using model predictive control. *Integrated Computer-Aided Engineering*, 22(2):171–185, 2015.

[185] Yang Zheng. *DMPC for platoons*, 2019. [Online]. Available: https://github.com/zhengy09/DMPC_for_platoons.

[186] Mohammad Hossein Basiri, Nasser L. Azad, and Sebastian Fischmeister. Secure dynamic nonlinear heterogeneous vehicle platooning: Denial-of-service cyber-attack case. In *Security and Privacy in Cyber-Physical Systems: Threats and Defenses. Studies in Systems, Decision and Control*. Springer, 2021.

[187] Mohammad Hossein Basiri, Benyamin Ghojogh, Nasser L. Azad, Sebastian Fischmeister, Fakhri Karray, and Mark Crowley. Distributed nonlinear model predictive control and metric learning for heterogeneous vehicle platooning with cut-in/cut-out maneuvers. In *59th Conference on Decision and Control (CDC), Jeju Island, Republic of Korea*. IEEE, 2020.

[188] Yang Zheng, Shengbo Eben Li, Jianqiang Wang, Dongpu Cao, and Keqiang Li. Stability and scalability of homogeneous vehicular platoon: Study on the influence of information flow topologies. *IEEE Transactions on intelligent transportation systems*, 17(1):14–26, 2016.

[189] Eric A Wan and Rudolph Van Der Merwe. The unscented kalman filter for nonlinear estimation. In *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No. 00EX373)*, pages 153–158. IEEE, 2000.

[190] Dan Simon. *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*. John Wiley & Sons, 2006.

[191] Yuan Yao, Lei Rao, Xue Liu, and Xingshe Zhou. Delay analysis and study of ieee 802.11 p based dsrc safety communication in a highway environment. In *2013 Proceedings IEEE INFOCOM*, pages 1591–1599. IEEE, 2013.

[192] Yunpeng Wang, Xuting Duan, Daxin Tian, Guangquan Lu, and Haiyang Yu. Throughput and delay limits of 802.11 p and its influence on highway capacity. *Procedia-Social and Behavioral Sciences*, 96:2096–2104, 2013.

[193] Xiaomin Ma, Xianbo Chen, and Hazem H Refai. Performance and reliability of dsrc vehicular safety communication: a formal analysis. *EURASIP Journal on Wireless Communications and Networking*, 2009:1–13, 2009.

[194] Vikas Kukshya and Hariharan Krishnan. Experimental measurements and modeling for vehicle-to-vehicle dedicated short range communication (dsrc) wireless channels. In *IEEE Vehicular Technology Conference*, pages 1–5. IEEE, 2006.

[195] Rami Sabouni and Roshdy M Hafez. Performance of dsrc for v2v communications in urban and highway environments. In *2012 25th IEEE Canadian Conference on Electrical and Computer Engineering (CCECE)*, pages 1–5. IEEE, 2012.

[196] P. W. H. Flannery B. P. and S. A. Teukolsky. *Numerical recipes. The art of scientific computing*. Cambridge University Press, 1986.

[197] Arpita Ghosh, Stephen Boyd, and Amin Saberi. Minimizing effective resistance of a graph. *SIAM review*, 50(1):37–66, 2008.

APPENDICES

Appendix A

Proofs of Chapter 3

A.1 Proof of Lemma 3.5

Before proving Lemma 3.5 we need the following preliminary definition.

Definition A.1 ([115])**.** A spanning subgraph of a graph $\mathcal{G}$ is called a 2-tree of $\mathcal{G}$, if and only if, it has two components each of which is a tree. In other words, a 2-tree of $\mathcal{G}$ consists of two trees with disjoint vertices which together span $\mathcal{G}$. One (or both) of the components may consist of an isolated node. We refer to $t_{ab,cd}$ as a 2-tree where vertices a and b are in one component of the 2-tree, and vertices c and d in the other. □

Based on the above definition, we prove Lemma 3.5.

Proof: The proof is based on the fact that $[L_g^{-1}]_{ij} = \frac{\text{cof}(L_g)_{ij,\ell,\ell}}{\det(L_g)}$. Thus, it is sufficient to provide graph-theoretic definitions of the nominator and denominator of this fraction. For the denominator, based on the generalization of matrix tree theorem for weighted graphs, we have

$$\det(L_g) = \prod_{i \in \mathcal{W}} w_i. \tag{A.1}$$

Moreover, for the nominator, the cofactor is equal to the sum of the impedance product of all 2-trees $t_{ij,\ell}$. Let us denote the set of edges in the path between nodes i and j by $\mathcal{R}_{ij}$. This path is unique since the graph is a tree. Defining $\mathcal{R}_{ij} = \{\mathcal{R}_{i\ell} \cup \mathcal{R}_{j\ell}\} \setminus \{\mathcal{R}_{i\ell} \cap \mathcal{R}_{j\ell}\}$ and $\pi_{\mathcal{R}} = \prod_{i \in \mathcal{R}_{ij}}$, we can write

$$\begin{aligned} \text{cof}(L_g)_{ij,\ell,\ell} = \pi_{\mathcal{R}} w_2 w_3 ... w_{|\mathcal{R}_{i\ell} \cap \mathcal{R}_{i\ell}|} + \pi_{\mathcal{R}} w_1 w_3 ... w_{|\mathcal{R}_{i\ell} \cap \mathcal{R}_{i\ell}|} \\ \pi_{\mathcal{R}} w_1 w_2 w_4 ... w_{|\mathcal{R}_{i\ell} \cap \mathcal{R}_{i\ell}|} + ... + \pi_{\mathcal{R}} w_2 ... w_{|\mathcal{R}_{i\ell} \cap \mathcal{R}_{i\ell}|-1}. \end{aligned} \tag{A.2}$$

By dividing (A.2) by $\det(L_g)$ from (A.1) the result will be obtained. ■

A.2 Proof of Lemma 3.6

Proof: Due to the triangular structure of L_g and L_g^{-1}, we can obtain the elements of L_g^{-1} by solving each row of $L_g^{-1}L_g = I$ in a recursive manner and the elements of L_g^{-1} will be obtained. ■

A.3 Proof of Theorem 3.14

Proof: Without loss of generality, we denote the ordering of the nodes in the path starting from the leader to the end of the path by $v_\ell, v_1, v_2, \ldots, v_j, \ldots, v_i, \ldots, v_n$. For the case that an extra edge is added between nodes j and i, using Sherman-Morrison formula [196] and (3.17) results in

$$\tilde{L}_g^{-1} = L_g^{-1} - \frac{w_i L_g^{-1}\mathbf{e}_{ij}\mathbf{e}_{ij}^{\mathsf{T}}L_g^{-1}}{1 + w_i\mathbf{e}_{ij}^{\mathsf{T}}L_g^{-1}\mathbf{e}_{ij}}. \tag{A.3}$$

Now we have

$$L_g^{-1}\mathbf{e}_{ij}\mathbf{e}_{ij}^{\mathsf{T}}L_g^{-1} = \begin{bmatrix} [L_g^{-1}]_{1i} - [L_g^{-1}]_{1j} \\ [L_g^{-1}]_{2i} - [L_g^{-1}]_{2j} \\ \vdots \\ [L_g^{-1}]_{ni} - [L_g^{-1}]_{nj} \end{bmatrix} \begin{bmatrix} [L_g^{-1}]_{i1} - [L_g^{-1}]_{j1} \\ [L_g^{-1}]_{i2} - [L_g^{-1}]_{j2} \\ \vdots \\ [L_g^{-1}]_{in} - [L_g^{-1}]_{jn} \end{bmatrix}^{\mathsf{T}} \tag{A.4}$$

where $\mathbf{e}_{ij} = \mathbf{e}_i - \mathbf{e}_j$. The diagonal elements of (A.4) have the form $\left([L_g^{-1}]_{ki} - [L_g^{-1}]_{kj}\right)^2$, $1 \leq k \leq n$ as the L_g^{-1} is a symmetric matrix. We need to show that the off-diagonal elements of (A.4) are non-negative. Without loss of generality let us suppose the form of the off-diagonal elements as $\left([L_g^{-1}]_{ki} - [L_g^{-1}]_{kj}\right)\left([L_g^{-1}]_{il} - [L_g^{-1}]_{jl}\right)$ for any $1 \leq k, l \leq n$. If $1 \leq k \leq j$ or $1 \leq l \leq j$, then $\left([L_g^{-1}]_{ki} - [L_g^{-1}]_{kj}\right)\left([L_g^{-1}]_{il} - [L_g^{-1}]_{jl}\right) = 0$ (based on Lemma 3.5). Let us suppose $j \leq k \leq i$. In this case one can easily verify that for any value of l, i.e., either $j \leq l \leq i$, or $i \leq l \leq n$, $[L_g^{-1}]_{ki} - [L_g^{-1}]_{kj}$ and $[L_g^{-1}]_{il} - [L_g^{-1}]_{jl}$ have the same sign. Now let us suppose $i \leq k \leq n$. With the same argument we conclude that $\left([L_g^{-1}]_{ki} - [L_g^{-1}]_{kj}\right)\left([L_g^{-1}]_{il} - [L_g^{-1}]_{jl}\right) \geq 0$ for any $j \leq l \leq n$. It is now sufficient to show that $w_i\mathbf{e}_{ij}^{\mathsf{T}}L_g^{-1}\mathbf{e}_{ij} \geq 0$. We have $\mathbf{e}_{ij}^{\mathsf{T}}L_g^{-1}\mathbf{e}_{ij} = [L_g^{-1}]_{ii} - 2[L_g^{-1}]_{ij} + [L_g^{-1}]_{jj}$ which is the effective

resistance of the added edge between nodes j and i and hence is positive [197]. Thus, the second term in (A.3) is a non-negative matrix. This implies that the elements of $\tilde{L}_g^{-1}$ are not larger than those of L_g^{-1}. This along with Lemma 3.4 complete the proof. ■

A.4 Proof of Theorem 3.15

Proof: Without loss of generality, we denote the ordering of the nodes in the path starting from the leader to the end of the path the same as proof of Theorem 3.14. For the case that an extra edge is added from node j to node i (not making a cycle), using Sherman-Morrison formula [196] and (3.18) results in

$$\tilde{L}_g^{-1} = L_g^{-1} - \frac{w_i L_g^{-1}\mathbf{e}_i\mathbf{e}_{ij}^{\mathsf{T}}L_g^{-1}}{1 + w_i\mathbf{e}_{ij}^{\mathsf{T}}L_g^{-1}\mathbf{e}_i}. \tag{A.5}$$

Now, we have

$$L_g^{-1}\mathbf{e}_i\mathbf{e}_{ij}^{\mathsf{T}}L_g^{-1} = \begin{bmatrix} & & & \overbrace{}^{j^{\text{th}}} & & \overbrace{}^{i^{\text{th}}} & & \\ \frac{1}{w_1} & 0 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \frac{1}{w_1} & \frac{1}{w_2} & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots & & \vdots & & \vdots \\ \frac{1}{w_1} & \frac{1}{w_2} & \cdots & \frac{1}{w_j} & \cdots & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots & & \vdots & & \vdots \\ \frac{1}{w_1} & \frac{1}{w_2} & \cdots & \frac{1}{w_j} & \cdots & \frac{1}{w_i} & \cdots & 0 \\ \vdots & \vdots & & \vdots & & \vdots & & \vdots \\ \frac{1}{w_1} & \frac{1}{w_2} & \cdots & \frac{1}{w_j} & \cdots & \frac{1}{w_i} & \cdots & \frac{1}{w_n} \end{bmatrix}$$

$$
\begin{aligned}
&\times \begin{bmatrix} 0 \\ 0 \\ \vdots \\ \vdots \\ \vdots \\ \underbrace{1}_{i^{\text{th}}} \\ \vdots \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ \vdots \\ \underbrace{-1}_{j^{\text{th}}} \\ \vdots \\ \underbrace{1}_{i^{\text{th}}} \\ \vdots \\ 0 \end{bmatrix}^{\mathsf{T}} \begin{bmatrix} \frac{1}{w_1} & 0 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \frac{1}{w_1} & \frac{1}{w_2} & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots & & \vdots & & \vdots \\ \frac{1}{w_1} & \frac{1}{w_2} & \cdots & \frac{1}{w_j} & \cdots & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots & & \vdots & & \vdots \\ \frac{1}{w_1} & \frac{1}{w_2} & \cdots & \frac{1}{w_j} & \cdots & \frac{1}{w_i} & \cdots & 0 \\ \vdots & \vdots & & \vdots & & \vdots & & \vdots \\ \frac{1}{w_1} & \frac{1}{w_2} & \cdots & \frac{1}{w_j} & \cdots & \frac{1}{w_i} & \cdots & \frac{1}{w_n} \end{bmatrix} \\
&= \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ \vdots \\ \underbrace{\frac{1}{w_i}}_{i^{\text{th}}} \\ \vdots \\ \frac{1}{w_i} \end{bmatrix} \begin{bmatrix} 0 & \cdots & \underbrace{0}_{j^{\text{th}}} & \frac{1}{w_{j+1}} & \frac{1}{w_{j+2}} & \cdots & \underbrace{\frac{1}{w_i}}_{i^{\text{th}}} & 0 & \cdots & 0 \end{bmatrix} \\
&= \begin{bmatrix} & & & \overbrace{}^{j^{\text{th}}} & \overbrace{}^{j+1^{\text{th}}} & \overbrace{}^{j+2^{\text{th}}} & & \overbrace{}^{i^{\text{th}}} & & \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots & & \vdots \\ 0 & 0 & \cdots & 0 & \frac{1}{w_i w_{j+1}} & \frac{1}{w_i w_{j+2}} & \cdots & \frac{1}{w_i^2} & \cdots & 0 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots & & \vdots \\ 0 & 0 & \cdots & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \end{bmatrix} \geq 0
\end{aligned}
\tag{A.6}
$$

Furthermore,

$$
w_i \mathbf{e}_{ij}^{\mathsf{T}} L_g^{-1} \mathbf{e}_i = w_i \left([L_g^{-1}]_{ii} - [L_g^{-1}]_{ji} \right) = w_i \left(\frac{1}{w_i} - 0 \right) = 1 \tag{A.7}
$$

Hence, the second term in (A.5) is a non-negative matrix. This implies that the elements of $\tilde{L}_g^{-1}$ are not larger than those of L_g^{-1}. This result along with Lemma 3.4 prove the claim.

For the case that an extra edge is added from node i to node j (making a cycle), with the same argument it can be easily shown that the second term in (A.5) is a non-positive matrix. This completes the proof. ■